PathOGiST: a novel method for clustering pathogen isolates by combining multiple genotyping signals

by

Mohsen Katebi

B.Sc., Sharif University of Technology, 2016

Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

in the
School of Computing Science
Faculty of Applied Science

SIMON FRASER UNIVERSITY
Spring 2020

Approval

Name:	**Mohsen Katebi**
Degree:	**Master of Science (Computing Science)**
Title:	**PathOGiST: a novel method for clustering pathogen isolates by combining multiple genotyping signals**
Examining Committee:	**Chair:** Keval Vora Associate Professor **Leonid Chindelevitch** Senior Supervisor Associate Professor **Cedric Chauve** Supervisor Professor **Kay C. Wiese** External Examiner Associate Professor
Date Defended:	**Nov 29, 2019**

Abstract

In this work, we study the problem of clustering bacterial isolates into epidemiologically related groups from next-generation sequencing data. Existing methods for this problem mainly use a single genotyping signal, and either use a distance-based method with a pre-specified number of clusters, or a phylogenetic tree-based method with a pre-specified threshold.

We propose PathOGiST, an open-source algorithmic framework for clustering bacterial isolates by leveraging multiple genotypic signals and calibrated thresholds. PathOGiST uses different genotypic signals, clusters the isolates based on these individual signals with correlation clustering, and combines the clusterings based on the individual signals with consensus clustering.

We implemented and tested PathOGiST on three different bacterial pathogens - *Escherichia coli*, *Yersinia pseudotuberculosis*, and *Mycobacterium tuberculosis* - and found that it outperforms most existing methods. We conclude by discussing how our framework can be extended and some of the challenges that remain to be addressed.

Keywords: Bacterial pathogens; whole-genome sequencing; correlation clustering; microbiology; public health

Dedication

To my inspiration, my biggest supporters, and my number one fans, my family.

Acknowledgements

This work was made possible thanks to the advice and contributions of many different people. First, I would like to thank my senior supervisor Dr. Leonid Chindelevitch for his support and supervision. I also would like to acknowledge my supervisor Dr. Cedric Chauve for his insightful comments and help. My thesis has benefited from guidances and help of Dr. Pedro Feijao, and I take the opportunity to thank him. I want to express my gratitude to the PathOGiST other team members Julius Booth, Mehrdad Mansouri, Sean La, Alex Sweeten, Reza Miraskarshahi, Matthew Nguyen, Johnathan Wong, and Dr. William Hsiao.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Partitioning the isolates of a bacterial pathogen into epidemiologically related groups is an important challenge in public health microbiology. Specifically, such a partitioning, which we will refer to as a *clustering*, can provide information on particularly transmissible strains (super-spreaders) and identify where an intervention such as active case finding may be particularly beneficial. In combination with additional metadata, such as geography or time of observation, such a clustering can also help identify rapidly growing groups (transmission hotspots), narrow down the potential origins of an outbreak (index case), and distinguish between recent and historical transmission.

This problem is related to, but distinct, from the problem of reconstructing transmission chains - in the latter, incompletely sampled data (missing links) as well as the presence of multiple strains within a patient (within-host heterogeneity) are a common challenge. They are not as much of an issue for clustering because in this case, no attempt is made at establishing the exact chain of transmission, only groups of isolates that may be part of such a chain. Since identifying the exact chain of transmission uniquely determines the clustering, but not vice versa, clustering is a more tractable problem, at least from an information-theoretic point of view.

The clustering problem can leverage a variety of genotypic signals. Historically, fairly coarse genotypes such as VNTR (variable-number of tandem repeats, i.e. the number of copies of a set of pre-specified repeated regions in a strain) [41], PFGE (pulsed field gel electrophoresis) [18] and MLST (multi-locus sequence type, i.e. the alleles at a small number of pre-specified housekeeping genes) [23] have been the predominant mode of genotyping bacterial pathogens. These low-resolution signals, which we may refer to as “fingerprints”, could lead to incorrectly clustered strains [1] since unrelated bacterial isolates may happen to share identical fingerprints. With the advent of next-generation sequencing (NGS) [22], new genotypic signals have become available. These include SNP (single-nucleotide polymorphism) profiles [8], which can be identified at the whole-genome scale, and also wgMLST (whole-genome multi-locus sequence type) [24], which contains the alleles at all of the known genes in the organism of interest.

The majority of existing approaches for clustering bacterial isolates use a single genotypic signal, typically one of the higher-resolution ones, in isolation [14]. However, in this paper we argue that the principled combination of both low-resolution as well as high-resolution genotypic signals may lead to the optimal results when performing clustering.

Methodologically, existing approaches fall into one of two categories. Some methods - including those inspired by and used in metagenomics [30] - use a pure distance-based approach, whereby a sequence similarity cutoff threshold is chosen, and any pair of sequences whose similarity exceeds it are considered to be in the same cluster, with a transitive closure operator applied to ensure the result is a valid partition (i.e. if x is close to y and y is close to z, then all three will be in the same cluster even if x is not close to z). Alternatively, such methods may simply apply a standard clustering method, such as hierarchical clustering, to the pairwise distance matrix; in this case, the number of clusters is typically specified in advance [6]. Other methods - which tend to be more computationally expensive - leverage a phylogenetic tree reconstructed from the data to define clusters [2, 13]. They also typically require a similarity threshold, but may be less sensitive to outlier isolates, i.e. isolates that do not look similar to any others or to homoplasy, i.e. convergent evolution.

The framework we propose here, called PathOGiST, innovates on existing methods in several key ways. First, it leverages multiple genotypic signals extracted from its input NGS data. They can be further subdivided according to granularity into coarse and fine signals; the former get penalized only for grouping together isolates with different genotypes, not for splitting isolates with similar genotypes, while the latter get penalized for both of these. Second, it is based on a distance threshold, but does not apply a transitive closure operator to the similarity graph, or require a pre-specified number of clusters. Instead, it makes use of the *correlation clustering* paradigm, which tries to minimize the number of pairs of distant isolates within clusters while minimizing the number of pairs of close isolates between clusters. Third, it can be calibrated to different bacterial pathogens and genotyping signals.

Our results demonstrate that, when applied to a selection of three bacterial pathogens with annotated datasets publicly available - *Escherichia coli*, *Yersinia pseudo-tuberculosis*, and *Mycobacterium tuberculosis* - PathOGiST performs with a higher accuracy than other existing methods, both in terms of its ARI (adjusted Rand index) as well as CP (cluster purity). Our paper establishes that the use of calibrated thresholds and multiple genotypic signals can lead to an accurate clustering of bacterial isolates for public health epidemiology.

In chapter 2, we start by briefly providing related background in bacterial genomics, discussing recent developments linked to the use of WGS data, and introducing the pre-processing and quality control methods used for our experiments. Then, we elaborate on the different genotypic signals that we used, namely Single Nucleotide Variations (SNV), Multilocus Sequence Typing (MLST), Copy Number Variations (CNV), Spoligotyping, and k-mer based distances. Finally, we mention the most recent related work on the problem of identifying transmission clusters from WGS data, while briefly presenting their main approaches and discussing their advantages and drawbacks.

In chapter 3, we explain our proposed approach for clustering using the correlation clustering concept. To begin with, we define a pairwise similarity measure based on pairwise distances obtained from genotyping sequences, given a distance threshold. Then, we use this similarity matrix to construct the graph for correlation clustering. Moreover, we propose two different algorithm to solve the correlation clustering optimization problem. First, an exact method that solves the ILP which is computationally expensive. Second, a parallel heuristic that approximates the solution quickly. Furthermore, we identify a consensus clustering, given different clusterings from each genotypic signals, by formulating the problem as another instance of correlation clustering which can be solved with the very same algorithms. Finally, we describe two measures we use to evaluate the performance of our clustering, i.e. cluster purity (CP), and adjusted Rand index (ARI).

In chapter 4, we show the results of our experiments on three published datasets of bacterial pathogens, i.e. *Escherichia coli*, *Mycobacterium tuberculosis*, and *Yersinia pseudotuberculosis*. We explain the details of experimental setup we used to run PathOGiST, comparing our two implementations, with respect to the running time, memory usage, and performance. Furthermore, our results are compared with the results from two most recent published tools for clustering infectious pathogens: Phydelity, and TreeCluster. The results show that PathOGiST significantly outperforms the two baselines with respect to ARI. However, when comparing the CP for *M. tuberculosis*, TreeCluster beats PathOGiST in a few cases.

My contribution to the PathOGiST project is mostly in the clustering part. Given pairwise distance matrices obtained from different genotyping signals, after exploring several approaches for clustering, I adopted and implemented the correlation clustering method. Finally, I designed and ran experiments using our three datasets and compared the results with two other baselines.

Chapter 2

Related Work

2.1 Background

Whole-genome sequencing (WGS) data has become more accessible and less expensive to generate. This has aided molecular epidemiology for rapidly detecting bacterial pathogens, as well as *in silico* molecular genotyping. Although multiple tools exist for genotyping, the question of which methods are more appropriate in each situation remains unanswered [38]. This lack of consensus can become a barrier for public health workers in the context of infectious disease outbreaks due to a shortage of *a priori* information regarding the pathogenic agent. Therefore, the need for a multi-criterion pipeline that addresses the limitations of a single genotyping method has become greater than ever.

Genotpying methods have several applications such as clustering bacterial samples into strains [28], and reconstructing a chain of transmission events [27]. These tasks are not trivial as bacteria evolve quickly and in a variety of ways. Different variability in bacterial genomes can be caused by point mutations [5], homologous recombination [21], insertions and deletions, and extrachromosomal elements such as plasmids. Moreover, the rate of mutation in different species varies widely. For example, *Mycobacterium tuberculosis* accumulates on average 0.5 SNP's per year [44], in contrast to *Helicobacter pylori* which gathers over 30 SNP's per year [19].

The widespread use of WGS data has paved the way for high resolution genotyping methods to be applied, namely single nucleotide variations (SNV's), and analyzing core genome MLST (cgMLST) schemes. Meehan *et al.* showed that the clustering based on SNV provides superior discriminatory power, and at the same time, reduces the rate of clustering and average timespan of transmission [27]. On the other hand, low resolution methods such as Mycobacterial Interspersed Repetitive Unit-Variable Number Tandem Repeats (MIRU-VNTR) tend to have higher rates of clustering with lower discriminatory power.

2.2 Preprocessing and Quality Control

NGSweep (`https://github.com/WGS-TB/NGSweep`) is a preprocessing tool for bacterial Next Generation Sequencing (NGS) data. In addition, it performs quality control to guarantee that only high quality data is used for downstream analysis. For our analysis, NGSweep, a unified interface of various preprocessing and quality control tools, is employed along with in-house procedures to eliminate outliers and preprocess FASTQ sequences. The three main steps of this pipeline are: i) the detection of outliers from the dataset; ii) the removal of adapters and contaminated reads in each sequence; iii) the aggregation of quality control reports for each sample.

In the first step, all samples that are not considered to originate from the organism selected by the user are removed from the analysis. This is an important step, since it is not uncommon for some samples to be either contaminated or mislabeled, which could negatively affect the downstream analysis. Alternatively, some samples could have very low depth of coverage, and they would also be removed.

In the second step, adapters and low quality base pairs are trimmed, and optionally reads flagged as contamination are also removed. In the third step, aggregated quality control report is generated, allowing the user to easily check any quality problem among the input samples.

2.3 Genotyping

Multiple genotypting methods have been used for further analysis in this work. Samples are genotyped after the preprocessing step.

2.3.1 Single Nucleotide Polymorphism

A single nucleotide polymorphism (SNP) is a variation at a specific position in the genome. For example, at a specific base position in a genome, the A nucleotide appears in the majority of individuals, but the position is occupied by C in others. We say that there is a SNP at this specific position with two alleles, C, and A. The total number of variations between samples has been used to infer evolutionary distances in many pathogen studies, e.g. Mycobacterium tuberculosis [15], Salmonella enterica [31] and Escherichia coli [17]. Several SNP calling pipelines are available, and Snippy (`https://github.com/tseemann/snippy`) is among the best-performing pipelines in our experience.

2.3.2 Multilocus Sequence Typing

Multi-Locus Sequence Typing (MLST) was first introduced by Maiden *et al.* in 1998 [23] as a technique for classifying bacterial isolates into strains, and it became one of the most common methods for pathogen outbreak surveillance [34]. Each isolate sample is characterized by a specific allelic profile, based on an existing MLST scheme composed of a database of known alleles for a selected set of loci. MLST schemes are available for many important pathogens. For example, in the pubMLST database [35], most MLST schemes composed of around 7 to 9 housekeeping genes. Such small MLST schemes follow the initial design introduced by Maiden *et al.* [24], given the sequencing technologies available at the time – mostly Sanger sequencing – and the need to accommodate specific pathogen evolutionary modes such as horizontal gene transfer [34].

With the availability of whole-genome sequencing (WGS) data, MLST schemes based on a larger set of genes began to be designed, namely *core genome* MLST (cgMLST), that consider the set of core genes shared by a group of related strains (generally a few hundred genes), and even *whole genome* MLST (wgMLST) schemes that rely on a set of thousands of genes, covering most of the loci of the target isolates [24]. These large MLST schemes greatly improve the low resolution of small MLST schemes. As a result, they have proven to be valuable typing methods in many studies, and they are becoming standard approaches for pathogen surveillance [7]. In this study, we used the recently published MentaLiST [9], an in-house *k*-mer based MLST caller designed specifically for handling large MLST schemes.

2.3.3 Copy Number Variations (CNV)

Variable-Number Tandem Repeats (VNTR) are genomic regions where a short sequence of DNA is repeated consecutively. Although a fixed set of VNTRs is typically identified for a given species, the copy number at each VNTR varies between individuals within a species. While VNTRs are found in both prokaryotic and eukaryotic genomes, the methodology called multi-locus VNTR analysis (MLVA) is widely used to distinguish different strains of bacteria, as well as cluster strains that might be epidemiologically related, and investigate evolutionary rates.

PRINCE (Processing Reads to Infer the Number of Copies via Estimation) [26] is an in-house software that is able to accurately estimate the copy number of a VNTR given the sequence of a single repeat unit and its flanking sequences, by computing a statistical approximation of the local coverage inside the repeat region. This approximation is then mapped to the copy number using a linear function whose parameters are fitted to simulated data.

Note that it is possible for PRINCE to output a positive real value for any one of the regions (e.g. 4.33, 2.01) despite the fact that in reality, an organism can only have a whole number of copies of a repeat; the value can be interpreted as the expected number of copies at that locus, with the fractional part of the measurement quantifying the uncertainty in PRINCE's prediction.

2.3.4 Spoligotyping

Spacer oligonucleotide typing (Spoligotyping) is a genotyping method for M. tuberculosis, based on the presence or absence of 43 spacer sequences between several repeat regions in the clustered regularly interspersed short palindromic repeats (CRISPR) locus, also known as the direct repeat (DR) locus. The patterns of presence and absence can be encoded as a 43-digit binary code, where 1 denotes the presence and 0 the absence for each spacer. This code can also be translated into a 15-digit numerical code called the *spoligotype*.

Spoligotyping is commonly based on wet-lab techniques, but recent methods have been proposed to find the spoligotype from WGS data. Among those, we used SpoTyping [46], one of the fastest methods available. SpoTyping works by blasting the 43 spacer sequences against a database generated using the sequence reads of a sample, and the presence or absence of spacer sequences is determined if the number of error-free and 1-error hits exceed a certain threshold.

2.3.5 k-mer based distances

k-mer based distances methods measures the genetic relatedness of two samples by decomposing their sequences into two sets of k-mers (subsequences with the length of k) and computing the differences between the two sets. The greater the difference, the less related the samples are. This method also bypasses the use of alignment and assembly techniques which enables the estimation of *de novo* genetic similarity free of reference-bias.

k-mer Weighted Inner Product (kWIP) is a software that uses k-mer based distances methods to rapidly estimate genetic similarity of samples produced in WGS experiments [29]. kWIP measures genetic similarity of samples by calculating the inner product across all sample pairs using vectors representative of the different k-mers and occurrences of these k-mers found in their respective samples, and then multiplying the inner product by a weight vector generated using the Shannon entropy on a vector of occurrence frequencies.

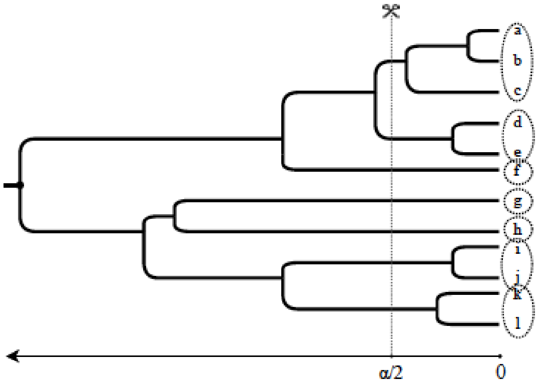
2.3.6 Distance matrices

For each genotyping method, a “distance” between pathogen samples is computed. kWIP outputs these distances outright, whereas distances based on CNVs, MLSTs, and SNVs are computed based on the vector representations of the genotypes. The hamming distance () is used for SNV, MLST, and spoligotyping, and the ℓ_1 norm is used for CNV.

2.4 Clustering

In the context of infectious disease, for identifying transmission clusters, phylogenetic approaches have been ubiquitous in the literature. This is mainly because phylogenies provide estimated distances between any two sequences in a sample and thus can be used for the clustering of genotyping data. The major drawback of these approaches is the computational burden of reconstructing phylogenetic trees, especially for large number of sequences. Furthermore, to estimate transmission clusters from an inferred phylogeny, a set of arbitrary criteria should be applied, and there is no consensus on how to do this in the literature. For example, most of them require a user-specified threshold or cutoff values, but do not describe an approach for selecting those values in different applications.

Figure 2.1: Phylogenetic clustering given an inferred ultrametric phylogeny and a threshold. Figure taken from [2]



Ragonnet-Cronin *et al.* in [36] introduced the Cluster Picker algorithm which clusters isolates given sequences, a phylogenetic tree, and a distance threshold. Clusters are defined as the leaves of a clade in the tree, where the number of clusters is minimized, and within-cluster pairwise sequence-based distances are below the threshold. This method scales cubically with the number of sequences.

Kosakovsky Pond *et al.* in [20] developed a tool called HIV-TRACE that when given sequences and a distance threshold, clusters isolates such that, a pair of isolates i and j are assigned to the same cluster if and only if the Tamura-Nei 93 (TN93) [40] distance between them is below the threshold. However, the computational complexity of the algorithm is quadratic with respect to number of sequences. Also, they apply a transitive closure operator to ensure the validity of the resulting clusters.

Villandré *et al.* in [42] proposed a Bayesian approach that aims to cluster HIV-1 sequences using phylogenetic distances between isolates. They implemented an algorithm called DM-PhyClus (Dirichlet-Multinomial Phylogenetic Clustering), that identifies groups of isolates resulting from quick transmission chains, thus yielding interpretable clusters, without using any arbitrary distance threshold. However, this approach is designed for viral genotypic sequences, especially HIV-1, and it tends not to perform well on bacterial sequences.

Han *et al.* in [13] developed a tool, called Phydelity, to infer putative transmission clusters, taking a phylogeny as input without requiring an arbitrary threshold. They use a patristic distance from phylogeny to find groups of sequences that are closely-related to each other. Still, it requires the number of clusters k, which also can be determined automatically by trying different thresholds in a given range and picking the best one with respect to a clustering criteria.

Balaban *et al.* in [2] proposed TreeCluster which takes phylogenetic trees as input, identifies clusters using theoretical algorithms for tree partitioning. They implemented a linear-time solution for the problem of HIV transmission clustering. We use this method for comparison purposes and will discuss the results further on.

Vrbik *et al.* in [43] proposed a distance-based clustering approach, called Gap Procedure, that avoids phylogenetic tree reconstruction and arbitrary threshold selection. Instead, they estimated pairwise distances from sequence alignments, then sorted them by size to identify larger 'gaps' between subsets. Although this procedure is quite fast, the uncertainty around its point estimates cannot be evaluated.

Chapter 3

Methods

The goal of our approach is to cluster pathogen isolates from whole-genome sequencing data by using different genotyping approaches, alone and in combination. Each cluster should ideally represent a set of isolates related by an epidemiological transmission chain. We assume that we are given as input several matrices recording the pairwise distances between the isolates, one per genotyping signal. The algorithm proceeds in two stages. We first compute a clustering of the isolates for each distance matrix, and then compute a consensus of these separate clusterings. For the first step, we rely on the notion of *correlation clustering* [3], which we describe in Section 3.1. For the second step, we use a modified approach to the consensus clustering problem [4], also based on a correlation clustering formulation, which we describe in Section 3.2. The whole process is illustrated in Figure 3.1.

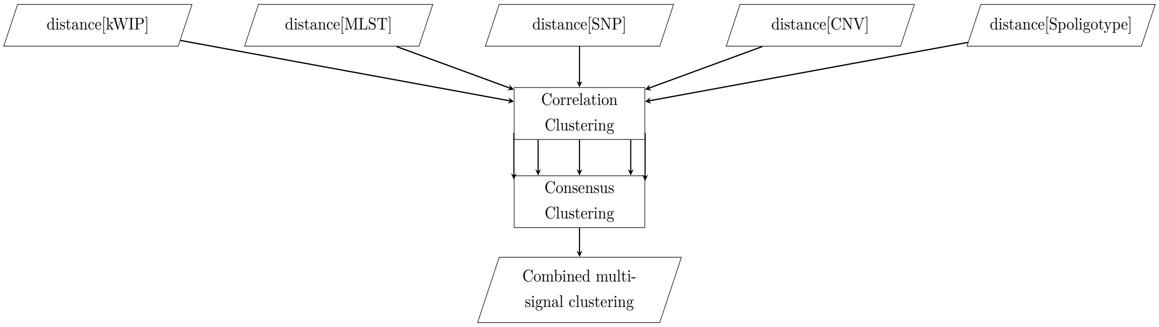


Figure 3.1: PathOGiST starts by computing clusters based on single distance signals using Correlation Clustering. Then we run Consensus Clustering on the outputs of the Correlation Clustering.

3.1 Correlation Clustering

Let G be an undirected, complete, weighted graph with vertices V and edges E. Let $W : E \to \mathbb{R}$ be an edge-weighting function, which is positive for vertices that are "similar" and negative for those that are "dissimilar". Correlation clustering aims to partition the vertices into disjoint clusters $C_1, C_2, \ldots, C_N$ where $N \leq n$. Let I be the set of edges whose endpoints lie in the same cluster and let $J = E - I$ be the set of edges whose endpoints lie in different clusters. The goal of correlation clustering is to minimize the sum of the weights of the edges in I with negative weight minus the sum of the weights of the edges in J with positive weight:

$$\min_{C_1, C_2, \ldots, C_N} \sum_{\substack{e_i \in I \\ W(e_i) < 0}}^{N} W(e_i) - \sum_{\substack{e_i \in J \\ W(e_i) > 0}}^{N} W(e_i)$$

In this work, we perform the construction of the weighted graph G from a distance matrix. Given a distance matrix D on the input elements (graph vertices), such that d_{ij} is the distance between elements i and j, we define $s_{ij} = T - d_{ij}$, where T is a *distance threshold*, intuitively meaning that if $s_{ij} > 0$, i and j are considered similar, while $s_{ij} < 0$ means that i and j are considered dissimilar. Thus, s_{ij} is the weight of the edge between vertices i and j in G.

The *minimum correlation clustering problem* aims to find a clustering that minimizes the sum of all positive s_{ij} for i, j in different clusters (penalty for separating similar pairs) minus the sum of all negative s_{ij} if i, j are in the same cluster (penalty for joining dissimilar pairs).

By defining binary variables x_{ij} such that $x_{ij} = 0$ if i and j are in the same cluster and $x_{ij} = 1$ otherwise, we can write the minimum correlation clustering objective function as

$$f(x) = \sum_{s_{ij} > 0} s_{ij} x_{ij} - \sum_{s_{ij} < 0} s_{ij}(1 - x_{ij}) = \sum s_{ij} x_{ij} - \sum_{s_{ij} < 0} s_{ij}.$$

Since the second term is constant, it follows that the minimun correlation clustering problem can be solved optimally with the following simple Integer Linear Program (ILP):

$$\underset{x}{\text{minimize}} \quad \sum s_{ij} x_{ij} \tag{3.1}$$

$$\begin{aligned} \text{s.t.} \qquad & x_{ik} \leq x_{ij} + x_{jk} \quad \text{for all } i, j, k \\ & x_{ij} \in \{0, 1\} \quad \text{for all } i, j \end{aligned}$$

Here, the inequality constraints (which we call the "triangle inequality" constraints) together with the binary constraints ensure that the assignment is transitive. Indeed, if $x_{ij} = 0$ and $x_{jk} = 0$, then i, j, k are all in the same cluster, which implies that $x_{ik} = 0$ [3].

3.1.1 C4: A fast parallel heuristic for the correlation clustering problem

Solving the ILP of Eq. (3.1) can be time consuming and can require a large amount of memory due to the quadratic number of variables and cubic number of constraints. For this reason, we additionally implemented the faster C4 algorithm, a parallel algorithm that guarantees a 3-approximation ratio of the optimal objective function of correlation clustering in the special case of metric distances (i.e. when the s_{ij} satisfy the triangle inequality $s_{ik} \leq s_{ij} + s_{jk}$) [33].

The idea of C4 is that given a graph G with positive and negative edges, in each step, a random vertex v is picked as a cluster center, and v and all its positive neighbors forms a cluster C_v. Then, all vertices in C_v are peeled from G and this process is repeated until all vertices are clustered. The process of picking a random vertex also is done in parallel with some concurrency control rules.

Our results show that this algorithm is remarkably fast and quite accurate on the input graphs we tested. However, it is non-deterministic, as it depends on the initial permutation of the vertices. With this in mind, we run C4 multiple times with random initial permutations and compute the objective value for each solution. Then, among those solutions, we choose the one that minimizes the objective function. Thus, multiple runs of C4 followed by the selection of the best one mitigate the non-deterministic nature of the algorithm. Our experiments show that in practice, this works very well and most of the time is able to find the optimum or near-optimum solution.

3.1.2 Solving the minimum correlation clustering problem exactly

In order to solve the minimum correlation clustering problem exactly, while coping with the $O(n^3)$ constraints in the ILP of Eq. (3.1), we employed two approaches.

First, recalling that we often get a near-optimum solution from C4, we use it as a warm start to the problem. In other words, we supply the solution produced by the C4 heuristic as a warm start to the ILP.

Second, rather than creating the ILP with all the constraints right away, we iteratively add the constraints as follows. According to Eq. (3.1), each constraint is in the form of the triangle inequality, i.e. for every set of three decision variables x_{ij}, x_{jk}, and x_{ik}, the ILP has three constraints

$$x_{ik} \leq x_{ij} + x_{jk},\ x_{ij} \leq x_{ik} + x_{jk},\ x_{jk} \leq x_{ik} + x_{ij}.$$

To give an intuition of our approach, assume all three similarities between elements i, j, k are positive. It implies that three elements i, j, and k are similar to each other, so are more likely to belong to the same cluster. In this case, the three variables x_{ik}, x_{ij}, and x_{jk} will be assigned value 0 and the three inequalities hold. On the other hand, all three similarities being negative implies that elements i, j, and k are likely to be in different clusters which would result again in the three inequalities holding.

Taking this into account, we use an approach inspired by constraint generation [10], and start by only including constraints induced by triplets of elements whose set of similarities contain both positive and negative edges and solve this trimmed-down ILP. We then check all the excluded constraints in the solution to see whether any of them is violated. If none is violated, then the current solution is also an optimum solution for the original ILP and we are done. Otherwise, we add all the constraints that are not satisfied by the current solution to the ILP and solve the modified ILP again. We repeat this process until no violated constraint remains.

In most experiments (225 out of 235 experiments), we observe that no violated constraints have been found. Almost all of the other cases only required one extra iteration to find a solution that satisfied all the constraints. The average number of iterations was 1.102.

3.2 Consensus Clustering

Given a set of clusterings and a measure of distance between clusterings, the *consensus clustering problem* aims to find a clustering minimizing the total distance to all input clusterings. A simple distance between two clusterings π_1 and π_2 is the number of elements clustered differently in π_1 and π_2, that is, the number of pairs of elements co-clustered in π_1 but not co-clustered in π_2, or vice versa.

Representing a clustering x by a quadratic number of binary variables ($x_{ij} = 0$ if and only if i, j are co-clustered), the distance between x and a clustering π is given by the formula

$$d(x, \pi) = \sum_{\pi_{ij}=1} w_{ij}(1 - x_{ij}) + \sum_{\pi_{ij}=0} w_{ij}x_{ij} = \sum_{i,j} s_{ij}x_{ij} + \sum_{x_{ij}=1} w_{ij}, \tag{3.2}$$

where a weight w_{ij} is assigned to each pair of elements i and j in the clustering x, and we defined $s_{ij} := (-1)^{\pi_{ij}} w_{ij}$.

Notice the connection with the minimum correlation clustering problem: solving the minimum consensus problem for a given set of clusterings $\pi^{(1)}, \ldots, \pi^{(n)}$ is equivalent to solving a minimum correlation clustering problem with the matrix S defined as

$$s_{ij} = \sum_{\left\{k | \pi_{ij}^{(k)}=0\right\}} w_{ij}^{(k)} - \sum_{\left\{k | \pi_{ij}^{(k)}=1\right\}} w_{ij}^{(k)} = \sum_{k=1}^{n} (-1)^{\pi_{ij}^{(k)}} w_{ij}^{(k)} \tag{3.3}$$

3.2.1 Consensus clustering with different granularities

An important feature of our problem is that the different genotyping signals we consider might not cluster the isolates with the same granularity. For example, it was shown in [27] that when clustering *Mycobacterium tuberculosis* isolates using SNPS, MLST, CNVs and spolygotypes, the latter two genotyping signals result in coarser clusters than SNPs and MLST. So we assume that the input clusterings can be of different granularities. In this setting, we want to avoid penalizing the differences between a finer clustering π and a coarser clustering π', and we introduce the following asymmetric distance: $d(\pi, \pi') = |\pi - \pi'|$. In this case, assuming x is the coarser clustering and π the finer one, we penalize only pairs that are co-clustered in π but not in π'.

Then, given the clusterings $\pi_1, \ldots, \pi_n$ and a subset F of these clusterings, representing the clusterings with finer resolution, the *finest consensus clustering* problem is to find a clustering x that minimizes the total distance between x and all input clusterings, where

$$d(x, \pi) = \begin{cases} \sum\limits_{\pi_{ij}=1} w_{ij}(1 - x_{ij}) + \sum\limits_{\pi_{ij}=0} w_{ij}x_{ij}, & \text{if } \pi \in F \\ \sum\limits_{\pi_{ij}=0} w_{ij}x_{ij}, & \text{otherwise} \end{cases} \tag{3.4}$$

We can then reformulate this problem as a minimum correlation clustering again, with matrix S defined by

$$s_{ij} = \sum_{\left\{k \mid \pi^{(k)}_{ij}=0\right\}} w^{(k)}_{ij} - \sum_{\left\{k \mid \pi^{(k)}_{ij}=1, \pi^{(k)} \in F\right\}} w^{(k)}_{ij} \tag{3.5}$$

3.2.2 Selecting appropriate weights for consensus clustering

There might be many meaningful ways of defining the weights $w^{(k)}_{ij}$ used in the previous equations. If we assume that a clustering π was inferred based on a distance matrix D, normalized such that $0 \leq d_{ij} \leq 1$, we can define w_{ij} as

$$w_{ij} = \begin{cases} d_{ij}, & \text{if } \pi_{ij} = 1 \\ 1 - d_{ij}, & \text{otherwise} \end{cases} \tag{3.6}$$

The reasoning behind this definition is that if $\pi_{ij} = 1$ (i, j are not co-clustered in π), then the distance d_{ij} should be large, therefore it is a good penalty for co-clustering i, j in x. On the other hand, if $\pi_{ij} = 0$, d_{ij} can be expected to be small, which means that $1 - d_{ij}$ is a better candidate for the penalty of choosing $x_{ij} = 1$. Therefore, the distance between two clusterings, given by Eq. (3.2), can be written as

$$d(x, \pi) = \sum_{\pi_{ij}=1} d_{ij}(1 - x_{ij}) + \sum_{\pi_{ij}=0} (1 - d_{ij})x_{ij} \tag{3.7}$$

and Eq. (3.3) becomes

$$s_{ij} = \sum_{\{k|\pi_{ij}^{(k)}=0\}} \left(1 - d_{ij}^{(k)}\right) - \sum_{\{k|\pi_{ij}^{(k)}=1\}} d_{ij}^{(k)} = \Pi_{ij} - D_{ij} \tag{3.8}$$

where $\Pi_{ij} = |\left\{k|\pi_{ij}^{(k)} = 0\right\}|$ and $D = \sum_{k=1}^{n} d_{ij}^{(k)}$.

We can naturally combine the weighting with the different granularities within a single formulation. In summary, the finest consensus clustering problem with weights can be formulated as a minimum correlation clustering problem, and thus solved by the algorithms described in Section 3.1.

3.3 Evaluation

To evaluate our methods for clustering, we compute two measures between our clustering and a ground truth clustering: Adjusted Rand Index (ARI) and Cluster Purity (CP).

The adjusted Rand index is a measure that computes how similar the clusters are to the ground truth. It is the corrected-for-chance version of the Rand index which is the percentage of correctly clustered elements. It can be computed using the following formula:

$$ARI = \frac{\sum_{ij} \binom{n_{ij}}{2} - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}\right] / \binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right] - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}\right] / \binom{n}{2}}$$

where n_{ij} ,a_i, b_j are values, row sums, and column sums from the contingency table [16].

Cluster Purity is another measure of similarity between two data clusterings. To compute, assign each cluster to the most common ground truth cluster in it. Then, count the number of correctly assigned data points and divide by the total number of data points. Formally:

$$CP(C, G) = \frac{1}{N} \sum_k \max_j |c_k \cap g_j|$$

where N is the number of data points, $C = \{c_1, c_2, \ldots, c_K\}$ is the set of clusters and $G = \{g_1, g_2, \ldots, g_J\}$ is the set of ground truth clusters [25].

Chapter 4

Results

4.1 Datasets and genotyping methods

We used three published datasets for three different pathogens, *Escherichia coli* [17], *Mycobacterium tuberculosis* [12], and *Yersinia pseudotuberculosis* [45]. Several genotyping signals were extracted from the WGS data: multilocus sequence typing (MLST) extracted by the in-house MentaLiST pipeline [9], single nucleotide polymorphisms (SNP) extracted by the open source Snippy tool [39], copy number variants (CNV) extracted by the in-house tool Prince [26], *k*-mer weighted inner products (kWIP) extracted by the open-source kWIP tool [29], and spacer oligonucleotide typing (Spoligotyping) performed by the open-source SpoTyping tool [46].

Table 4.1: Datasets and genotyping summary

Dataset	Number of isolates	Genotyping signals				
		SNP	MLST	kWIP	CNV	SpoTyping
E. coli	1509	✓	✓	✓	✗	✗
M. tuberculosis	1377	✓	✓	✓	✓	✓
Y. pseudotuberculosis	163	✓	✓	✓	✗	✗

For each genotyping signal, in order to apply our correlation clustering algorithm, we needed to determine a threshold T to decide which pairs of isolates should be considered similar. To do so, we consider the pairwise distance distribution for each signal, choosing a threshold range that covers the first valley in the distribution, under the assumption that the first peak likely indicates distances between isolates belonging to the same cluster. The resulting threshold ranges and spacings are described in Table 4.2.

4.2 Single signal genotyping

The largest dataset that we considered consists of 1509 *E. coli* isolates [17]. The dataset was collected from across England and spans an 11-year period. Samples are associated

Table 4.2: Genotyping threshold ranges (MTB is for *M. tuberculosis* and Yp for *Y. pseudotuberculosis*).

Dataset	SNP		MLST		kWIP		CNV		SpoTyping	
	range	step	range	step	range	step	range	step	range	step
E. coli	$(0, 43000]$	2150	$(0, 600]$	20	$[0.21, 0.75]$	0.03	-	-	-	-
MTB	$(0, 500]$	25	$(0, 500]$	25	$[0.125, 0.5]$	0.025	$(0, 50]$	2.5	$(0, 13]$	0.65
Yp	$(0, 40000]$	2000	$(0, 600]$	20	$[0.175, 0.7]$	0.025	-	-	-	-

with bloodstream infections, and were clustered using hierBAPS [6]. All isolates underwent whole-genome sequencing using the Illumina HiSeq 2000 sequencer. We considered three genotyping signals for this dataset: MLST, SNP, and kWIP. Then, for each signal and each threshold, we ran our two algorithms for solving the minimum correlation clustering problem, the C4 approximation algorithm (with multiple runs) and the exact ILP using delayed constraint generation. The distance distribution for each genotyping signal and the ARI of the resulting clusterings compared to the clustering provided in [17], used as the gold standard, is shown in Fig. 4.1, 4.2, and 4.3.

For our second dataset, we used a subset of 163 isolates of *Y. pseudotuberculosis* mostly collected from New Zealand, as described in [45], and sequenced using the Illumina NextSeq sequencer. We applied the same genotyping method as for the *E. coli* dataset. The results are presented in Fig. 4.4, 4.5, and 4.6.

The third dataset we considered is the *M. tuberculosis* dataset obtained described in [12]. All the isolates are from pediatric patients in British Columbia, Canada, and were collected between 2005 and 2014. We used a subset of 1377 isolates, all of which underwent WGS. In addition to SNP, MLST and kWIP information, we considered two additional genotyping signals, CNV and Spolygotyping. The results for this dataset are illustrated in Fig. 4.7, 4.8, 4.10, 4.9, and 4.11.

For *E. coli* and *Y. pseudotuberculosis*, we consider the MLST groups determined in their respective studies [17, 45] as the ground truth, and use them to calculate the ARI and CP values. For *M. tuberculosis*, due to the lack of MLST groups, we use the strain's lineage, a proxy for its geographic origin [37]. For this dataset, the ARI would not be informative since a lineage is a coarse grouping largely uninformative of the underlying epidemiology, and should be split into multiple clusters. Thus, we only calculate and report the CP in this case.

We can observe that for all the pathogens and genotyping signals we consider, there is a relatively clear threshold that falls within the chosen range which results in high accuracy clusters, with ARI and CP values above 0.8, and often close to 1. The only exceptions concern the *M. tuberculosis* dataset with SNPs, MLST and kWIP. Moreover, most of the time, around the best thresholds, the clustering obtained with the exact ILP method results in accuracy measures that are either very close to those of the C4 method or slightly better. Again, the most notable differences occur with the *M. tuberculosis* dataset with the SNP, MLST and Spolygotyping signals, where the exact ILP method often performs much better than the C4 method. These results illustrate the need to consider various genotyping signals for clustering pathogen isolates.

Figure 4.1: Distance histograms and ARI results for SNP, *E. coli.*

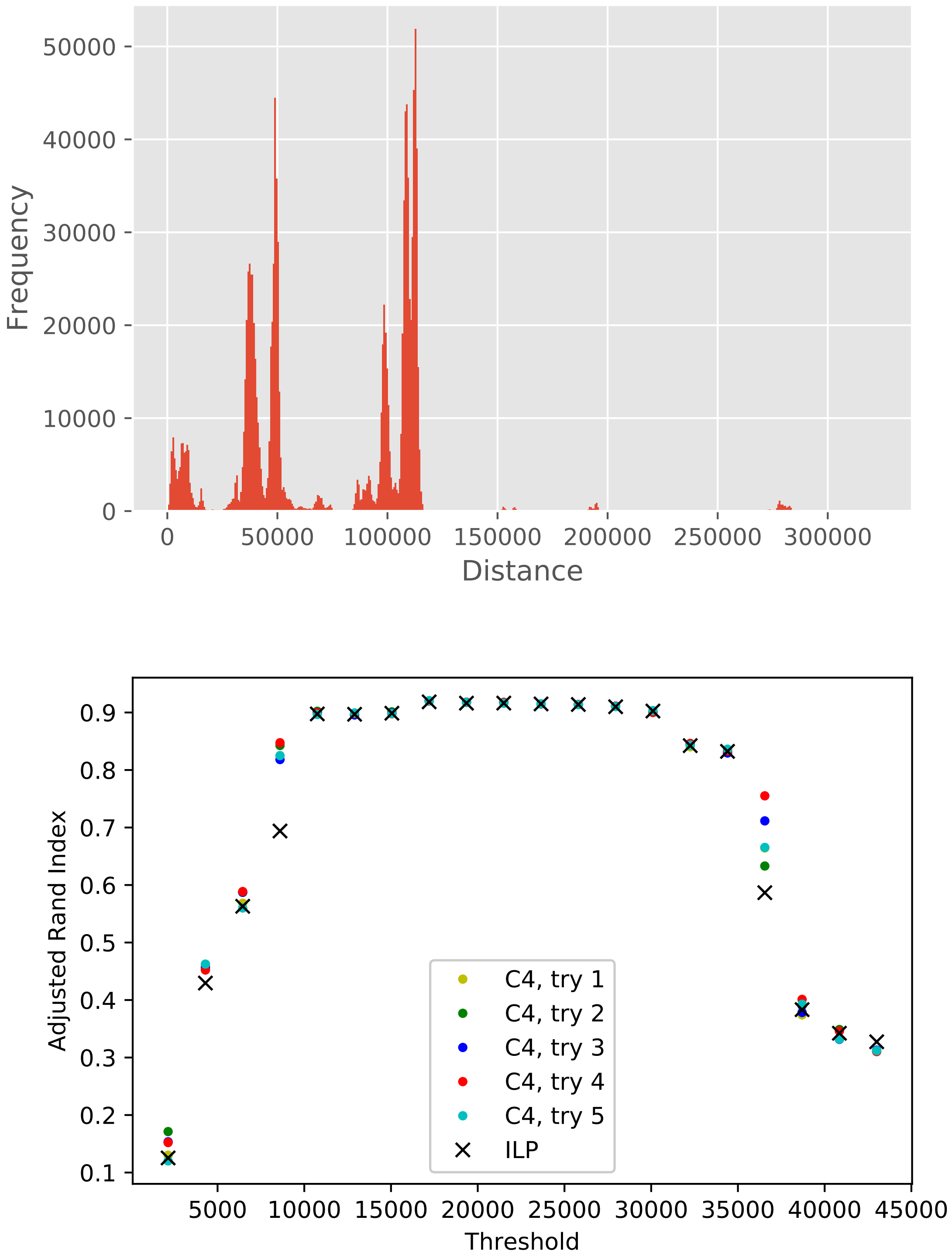
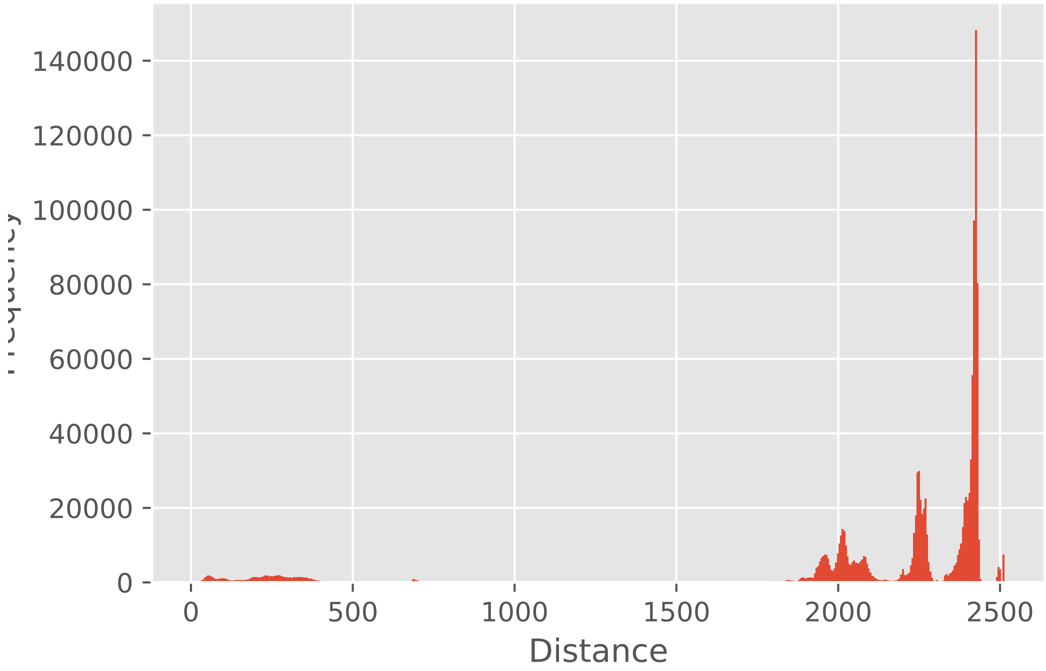


Figure 4.2: Distance histograms and ARI results for MLST, *E. coli*.



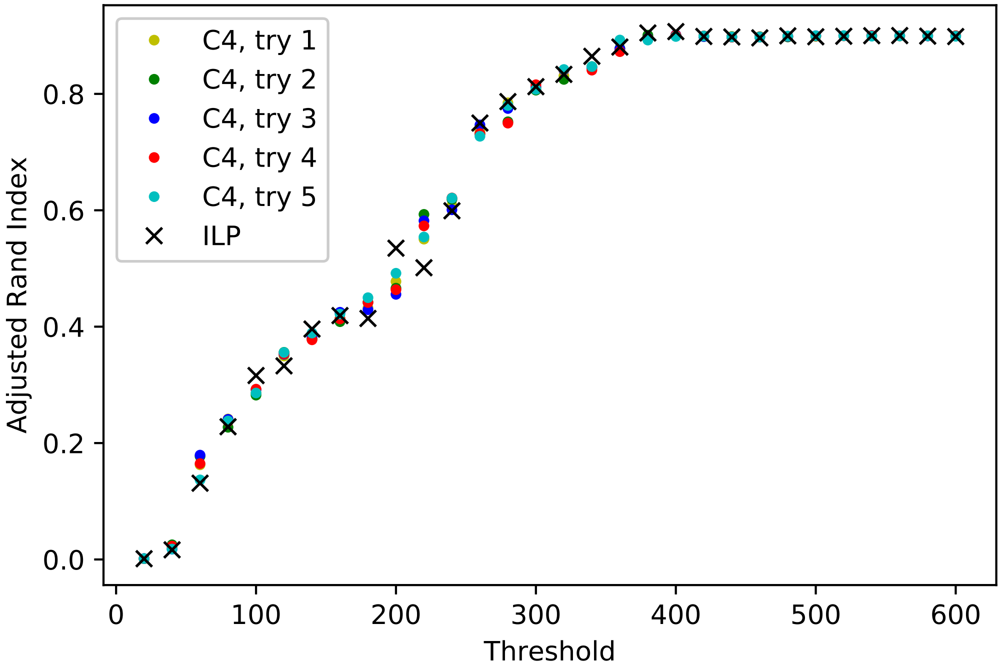


Figure 4.3: Distance histograms and ARI results for kWIP, *E. coli*.

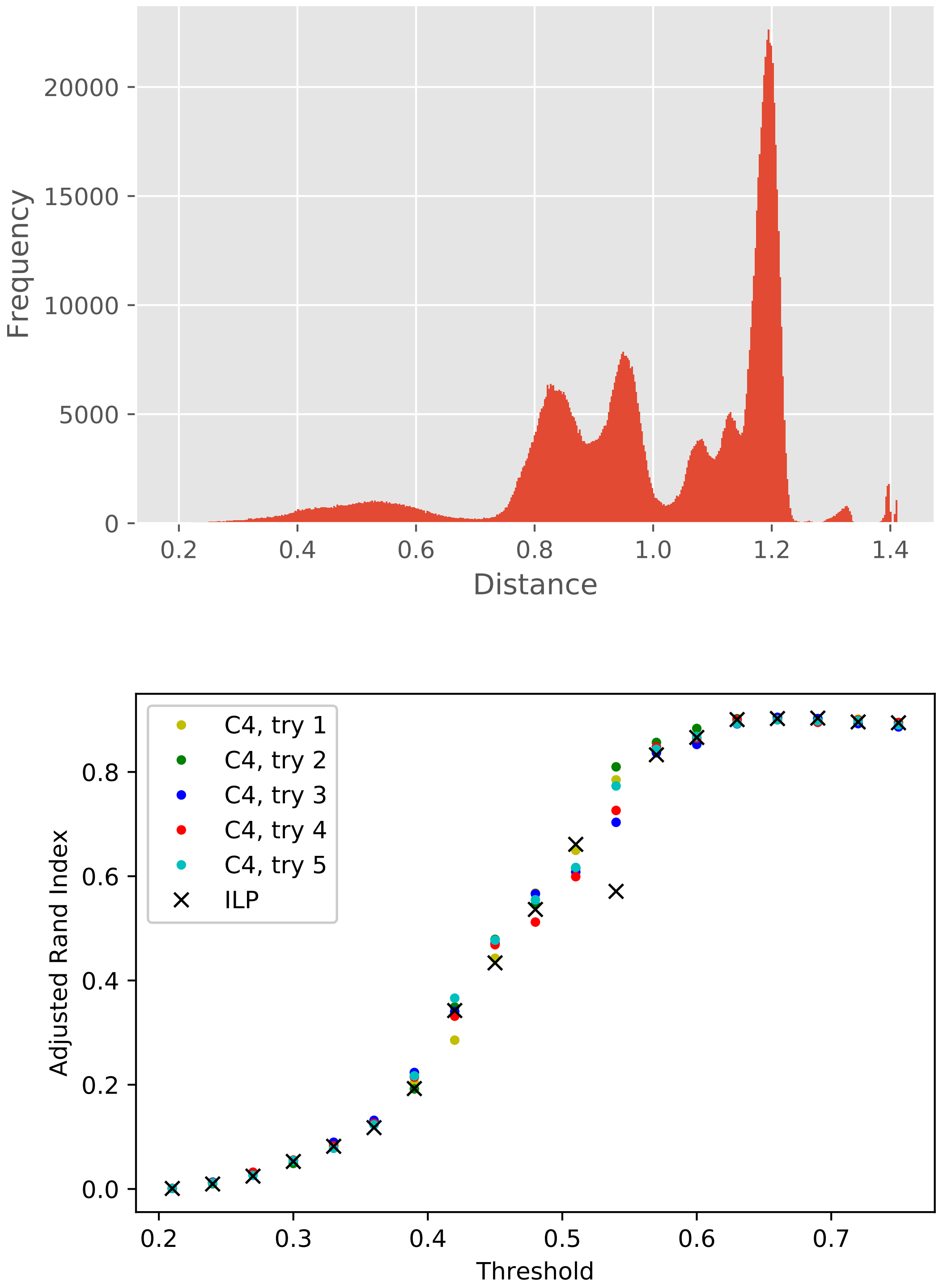


Figure 4.4: Distance histograms and ARI results for SNP, *Y. pseudotuberculosis*.

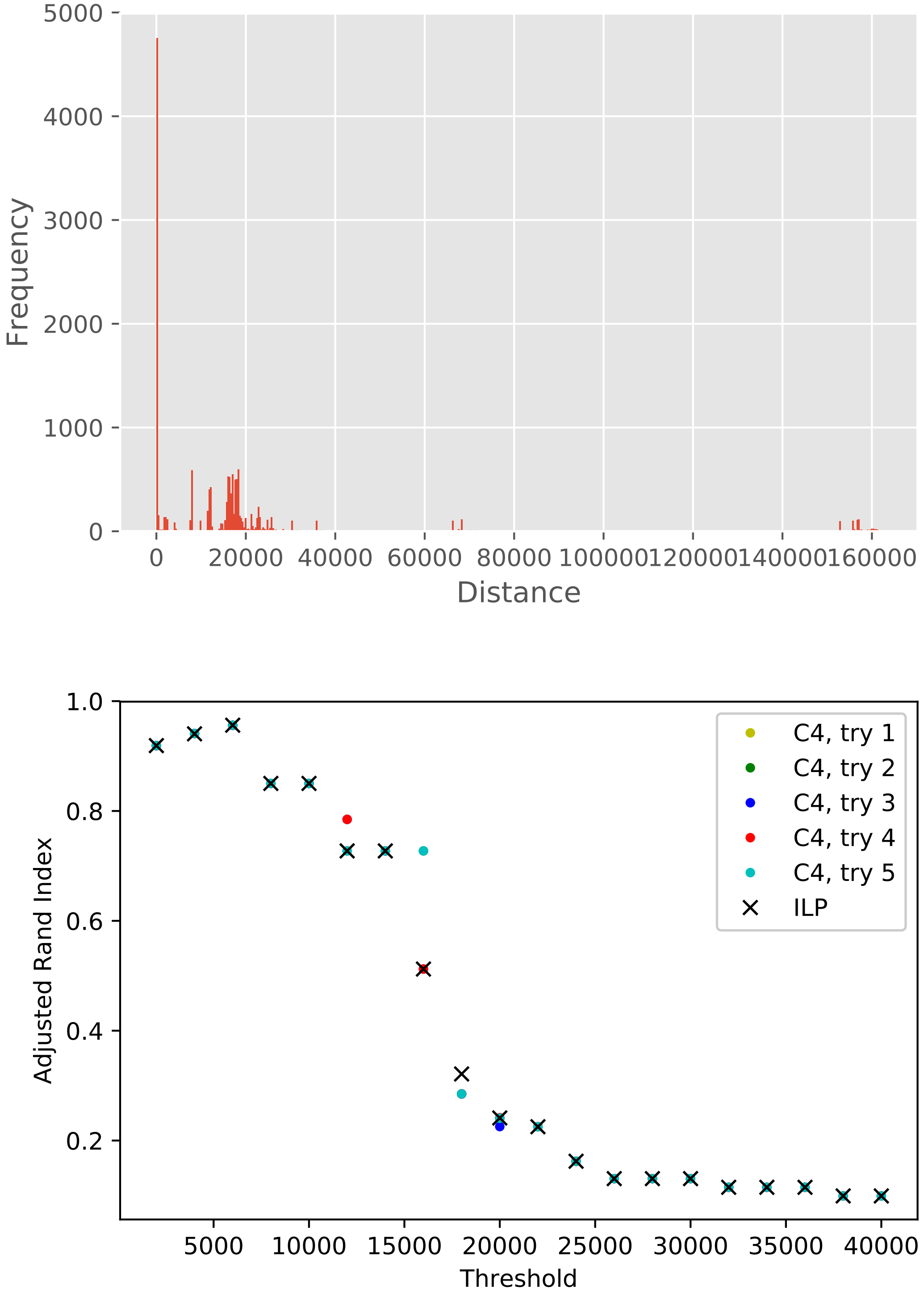


Figure 4.5: Distance histograms and ARI results for MLST, *Y. pseudotuberculosis*.

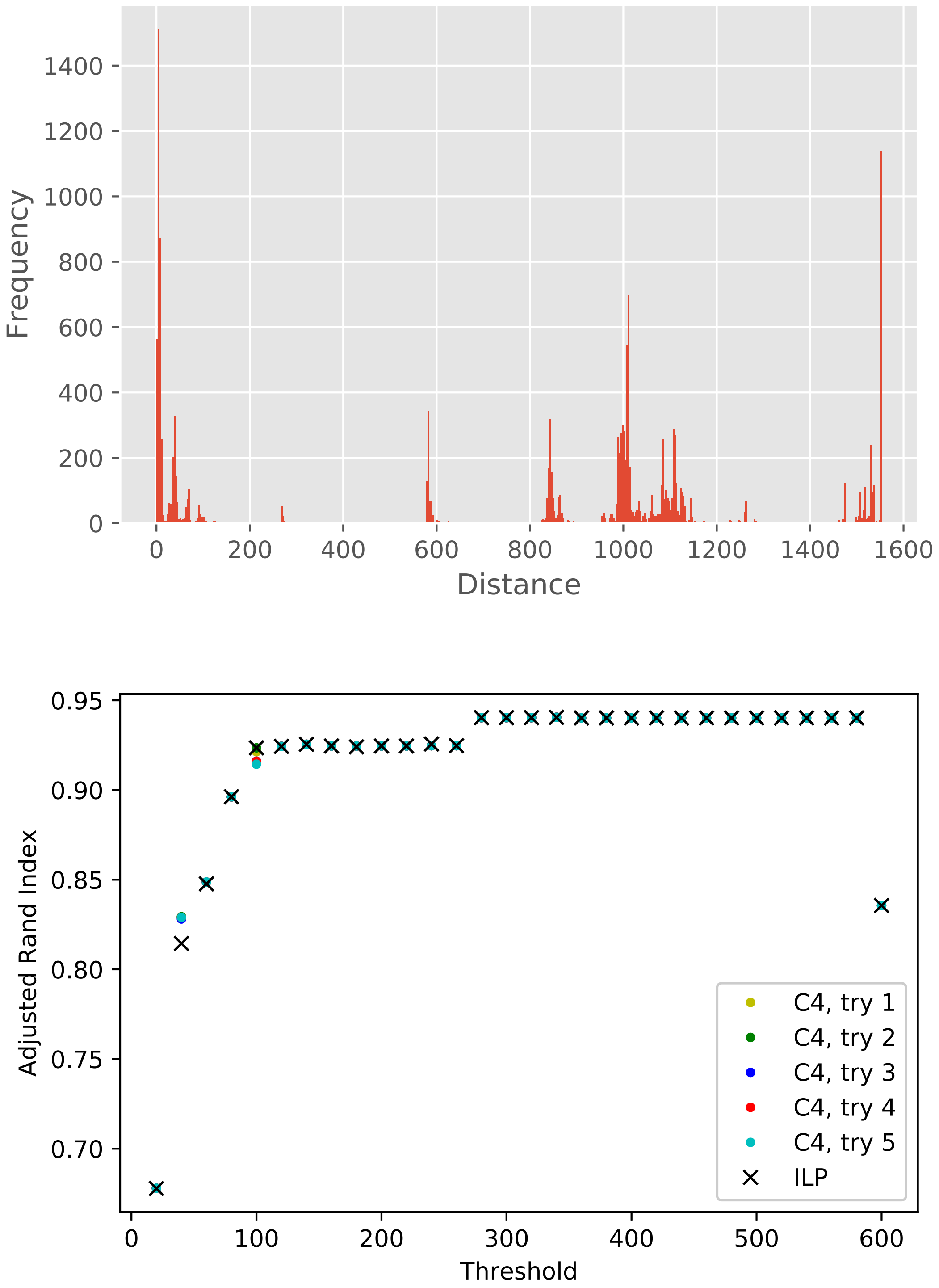


Figure 4.6: Distance histograms and ARI results for kWIP, *Y. pseudotuberculosis.*

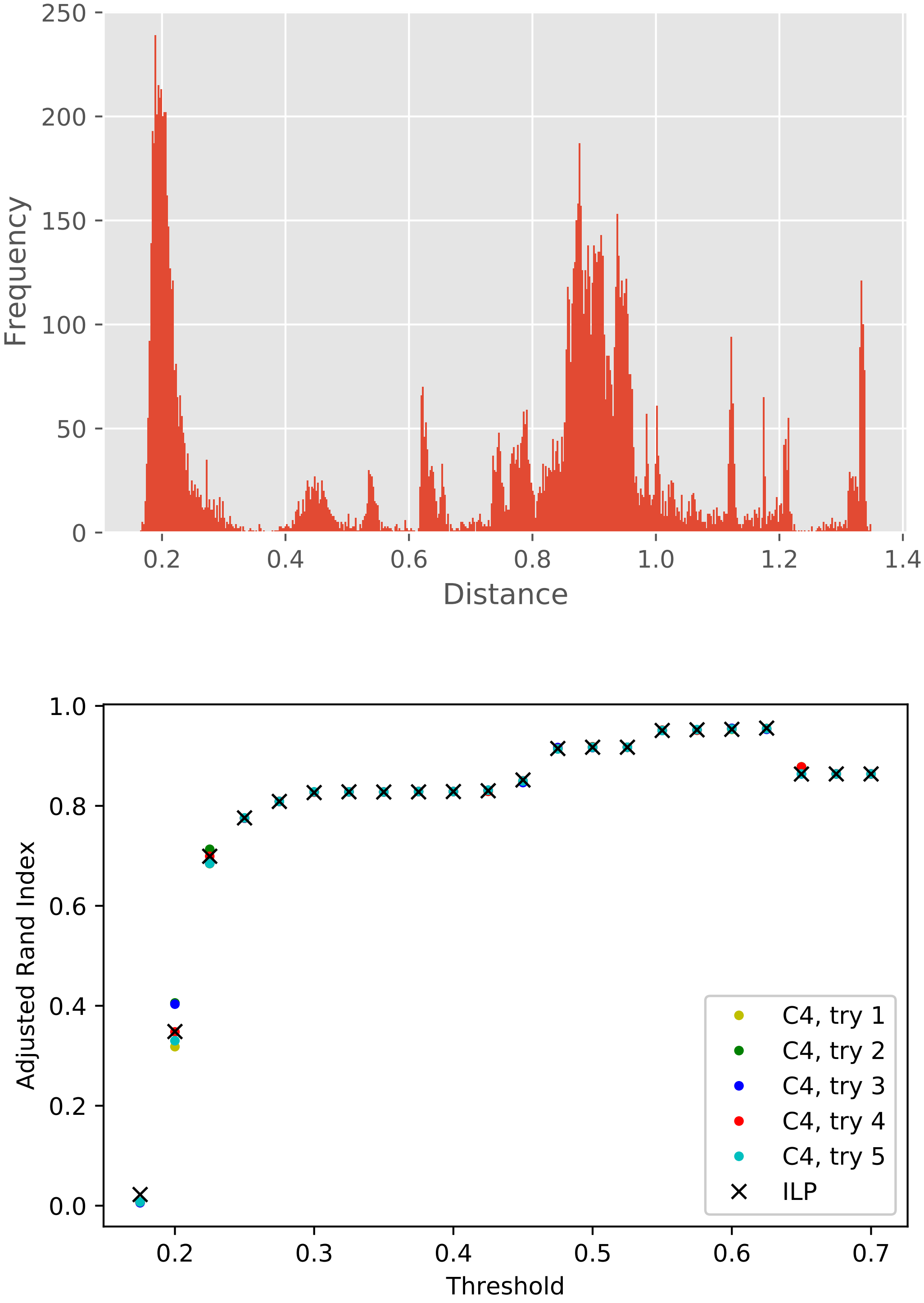


Figure 4.7: Distance histograms and CP results for SNP, *M. tuberculosis*.

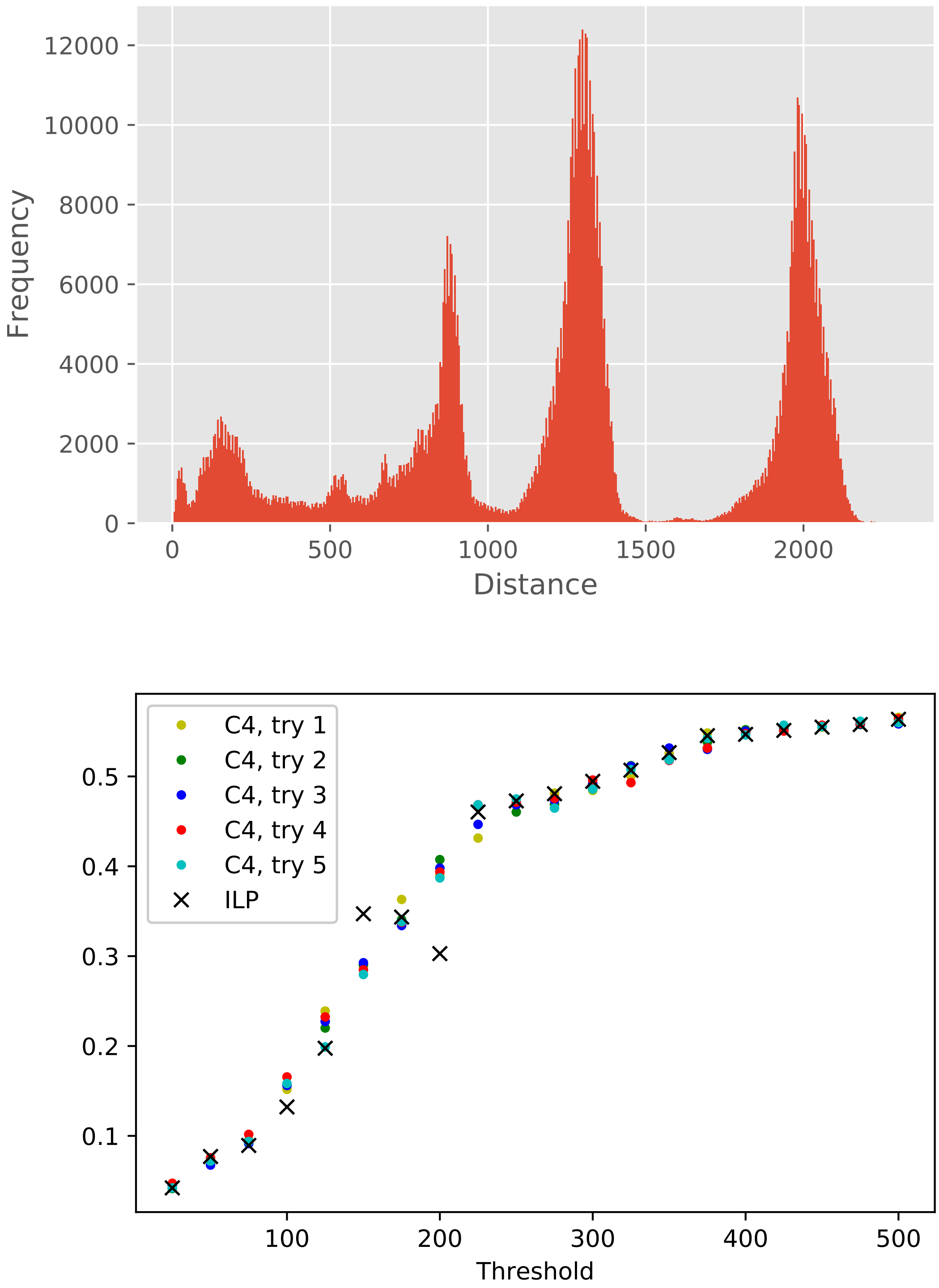
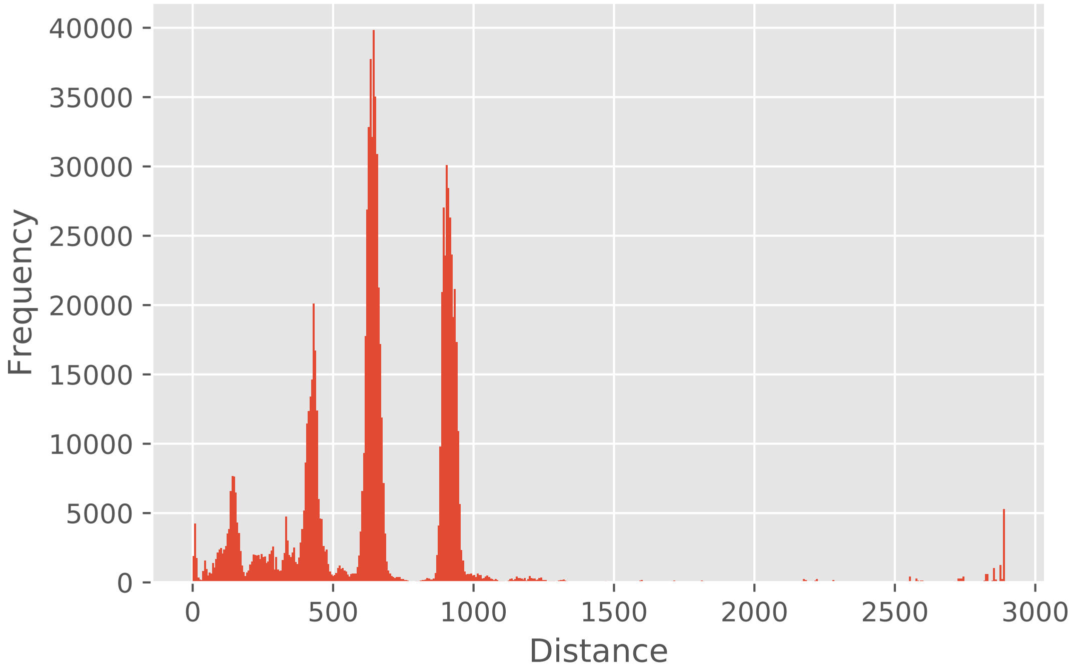


Figure 4.8: Distance histograms and CP results for MLST, *M. tuberculosis*.



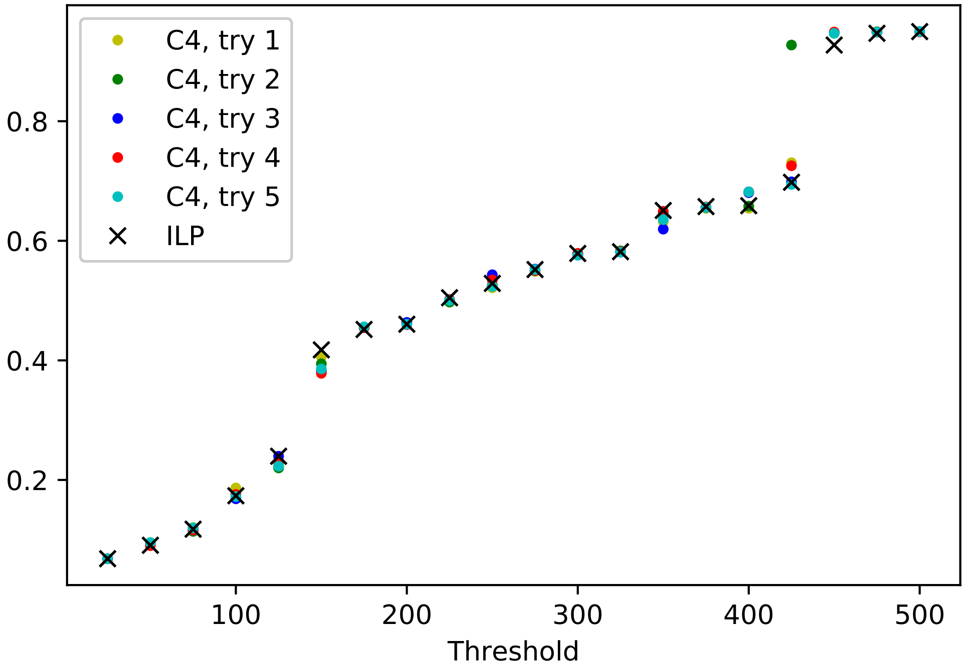


Figure 4.9: Distance histograms and CP results for kWIP, *M. tuberculosis*.

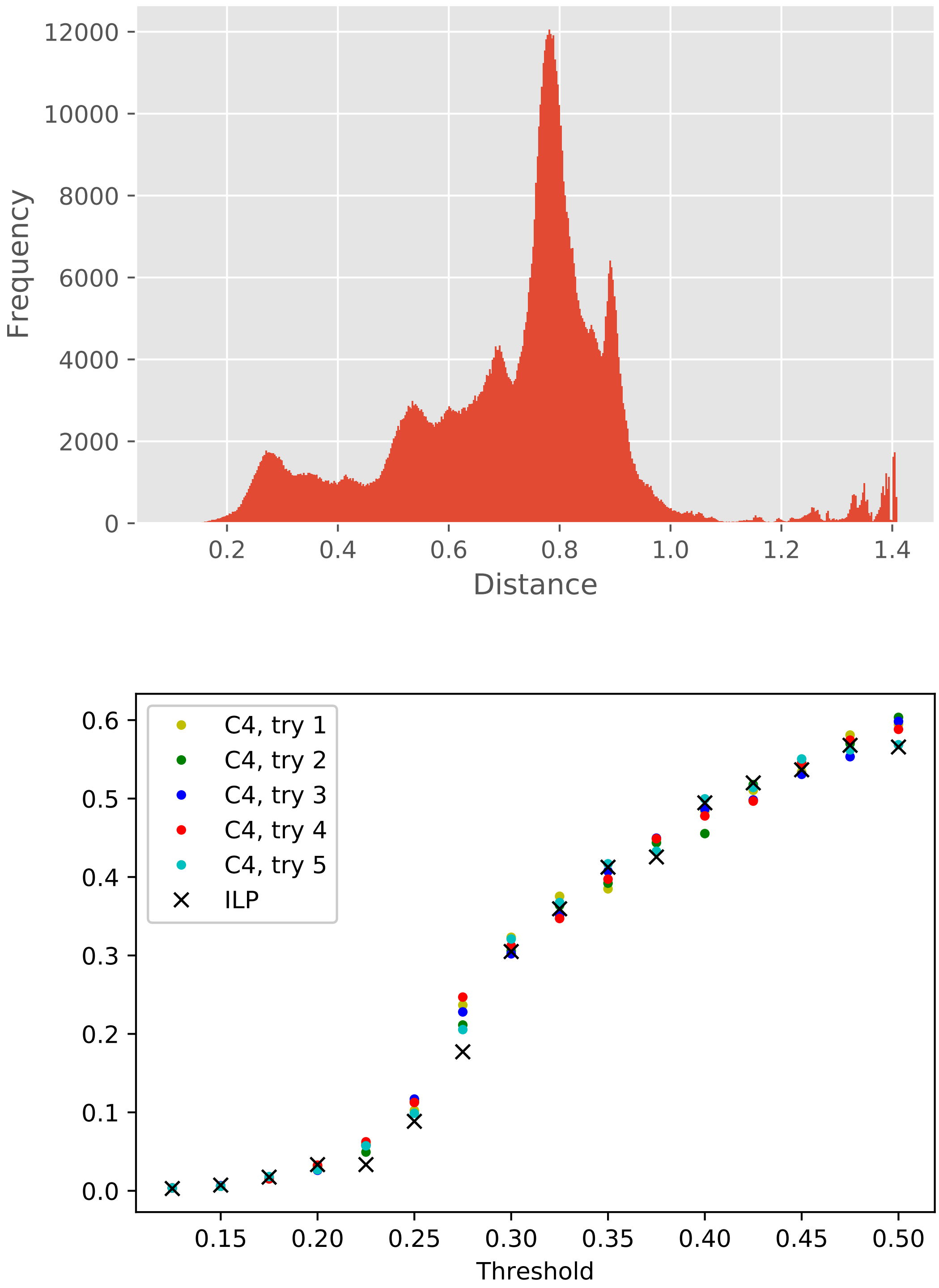


Figure 4.10: Distance histograms and CP results for CNV, *M. tuberculosis*.

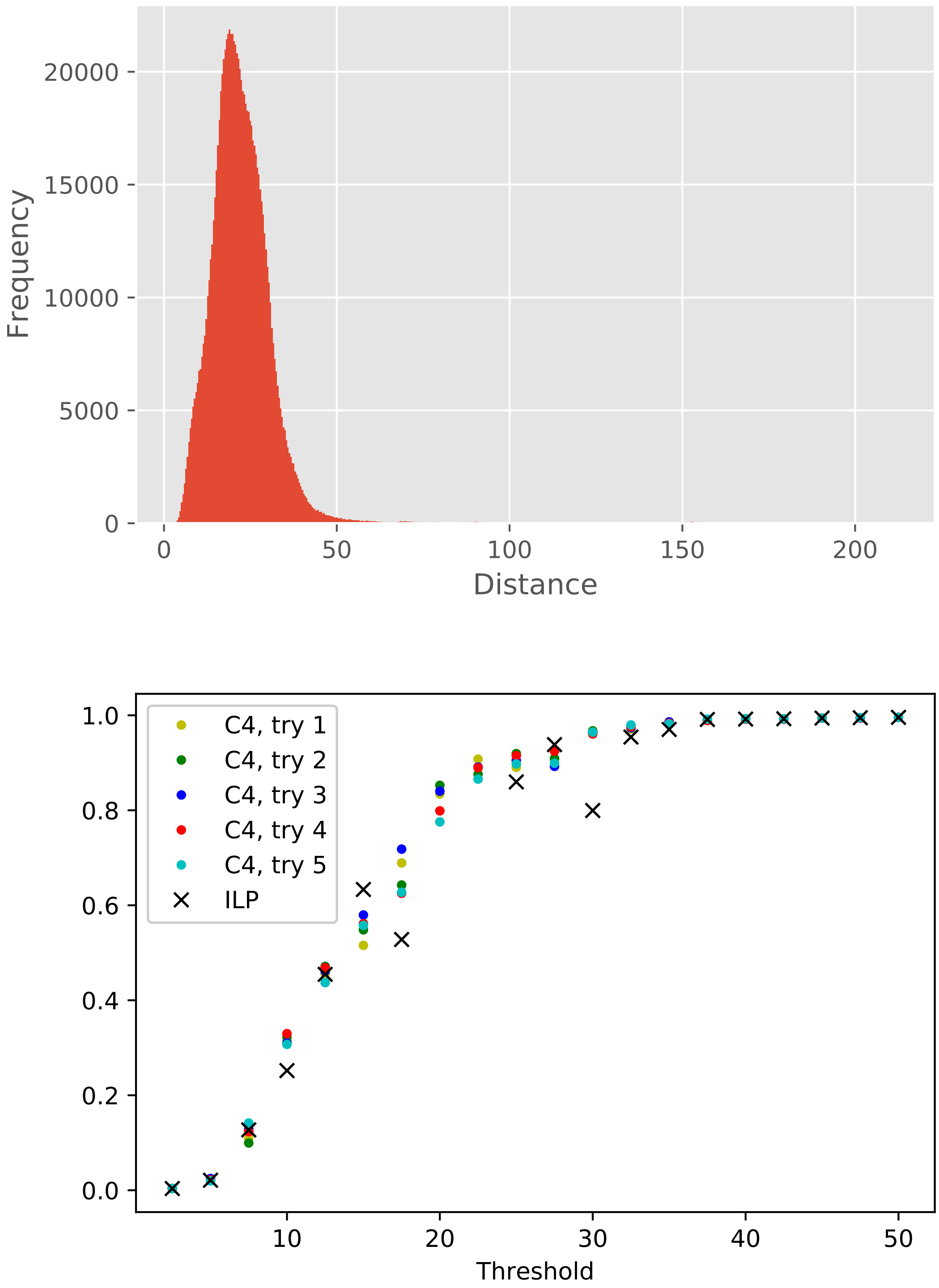
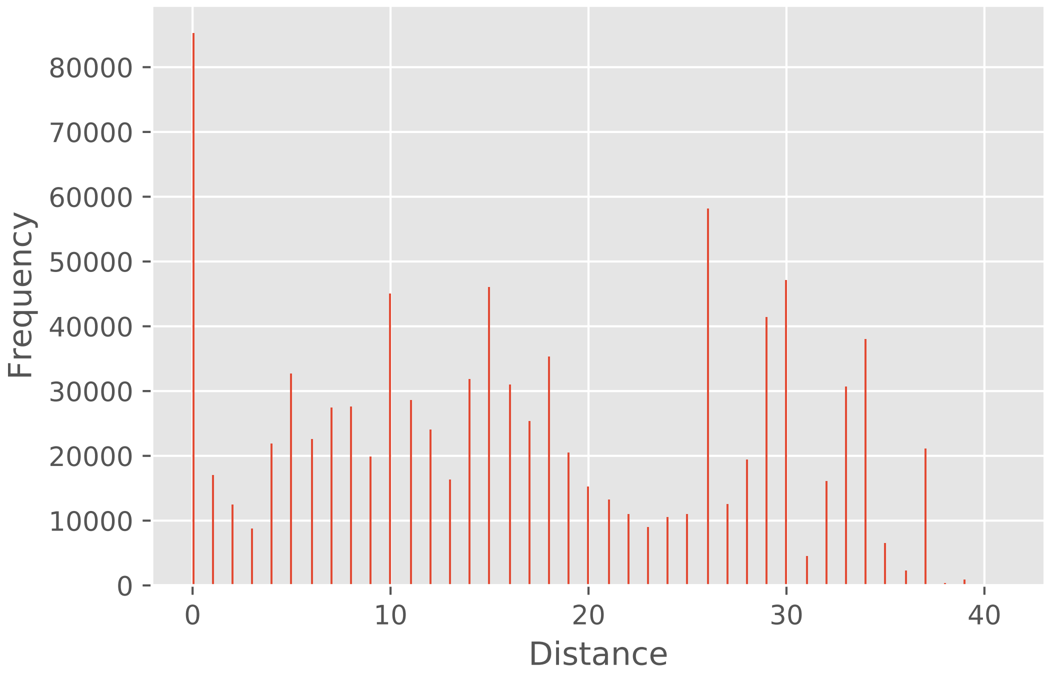
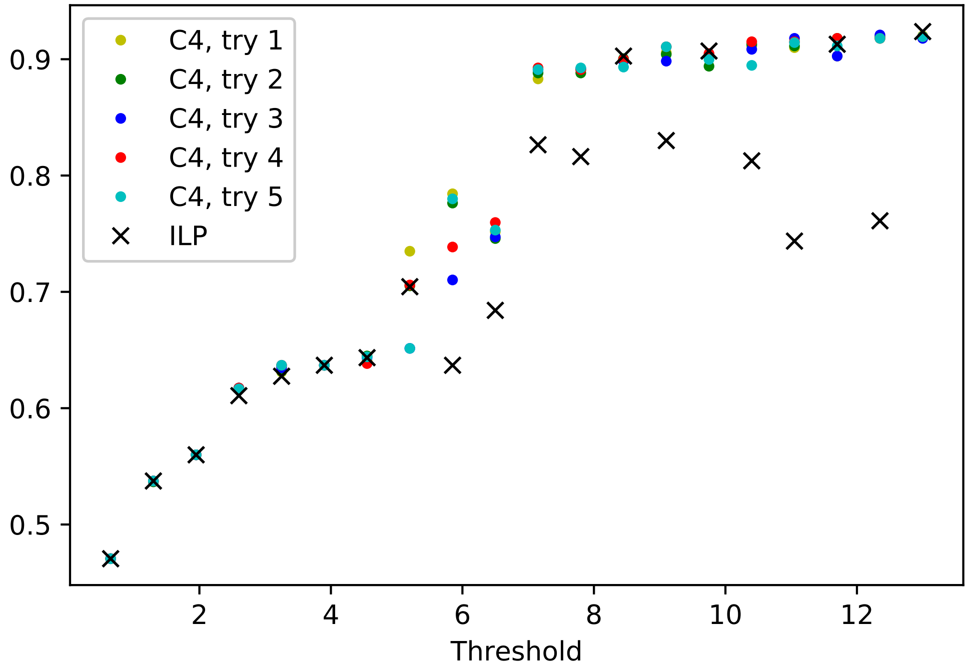


Figure 4.11: Distance histograms and CP results for Spoligotyping, *M. tuberculosis*.





4.3 Comparison of the C4 and exact ILP methods

The ILP generally gives more accurate results and is a deterministic method. However, its running time and memory usage depend a lot on the size of the dataset. For example, while it is able to cluster the smaller *Y. pseudotuberculosis* dataset in less than a minute, it takes more than three hours to find clusters for larger datasets at some threshold values. On the other hand, the C4 heuristic is significantly faster and requires much less memory even on the larger datasets, as shown in table 4.3. However, it is not deterministic, and random restarts may give slightly different, incompatible results. To evaluate the C4 heuristic performance, we compared the objective values of solutions found by C4 and by the exact ILP. In most cases, C4 performs very well and finds a solution whose objective value is close to the optimal (Table. 4.4). Furthermore, the objective value of CPLEX is affected by the tolerance parameter; when the gap between the lower and upper bound is less than a certain fraction ϵ, set to 10^{-6} by default, the optimization is stopped. In this case, we see that because the magnitude of the objective function is fairly large, it is possible for the C4 method to obtain a better objective function than CPLEX. However, this can be modified by adjusting the tolerance. One challenge for running exact ILP for larger datasets is that it requires a massive amount of memory. To overcome this challenge, we ran our experiments on machines with 1TB of memory.

Table 4.3: Average running time (in seconds) and memory footprint (in gigabytes).

Dataset	Time (s)		Memory (GB)	
	C4	ILP	C4	ILP
E. coli	698	7282	0.22	193.72
M. tuberculosis	572	10437	0.20	298.87
Y. pseudotuberculosis	14	15	0.13	0.81

Table 4.4: ILP and C4 objective value comparison

Dataset	ILP	C4: mean	C4: std
E. coli	-1.9068×10^{10}	-1.9074×10^{10}	3.8817×10^{5}
M. tuberculosis	-2.9445×10^{8}	-2.8844×10^{8}	2.4238×10^{3}
Y. pseudotuberculosis	-4.1601×10^{7}	-4.1594×10^{7}	1.3238×10^{3}

4.4 Comparison with existing clustering methods

The results from PathOGiST were compared to those generated by two recent methods developed for clustering WGS datasets, Phydelity [13] and TreeCluster [2]; both of them are based on phylogenetic trees. To infer a phylogeny for our datasets, we first calculated a pair-wise distance matrix using Mash [32], then ran the popular and widely used BIONJ [11] variant of the neighbor joining algorithm on the distance matrix. After Inferring these phylogenetic trees, we ran Phydelity and TreeCluster with their default settings. In order to pick a single threshold for each genotyping signal-pathogen combination in PathOGiST, we chose the threshold resulting in the best ARI (CP for *M. tuberculosis*) among all the options. These thresholds are set as the default thresholds for these genotyping signal-pathogen combinations, but can be overriden by the user. Table 4.5 shows the chosen optimal threshold for each dataset and genotyping signal.

Table 4.5: Best clustering thresholds per dataset and genotyping signal.

Dataset	SNP	MLST	kWIP	CNV	SpoTyping
E. coli	17200	400	0.66	-	-
M. tuberculosis	500	475	0.5	50	13
Y. pseudotuberculosis	6000	340	0.625	-	-

Having clustering outputs of the single signal correlation clustering algorithm with chosen default thresholds, we ran consensus clustering for each pathogen with all their available genotyping signals, considering SNP clustering as the finest. The results are described in Table 4.6. The main observation is that in all cases, but *M. tuberculosis*, the consensus clustering ARI is close to the best ARI obtained by a single genotyping signal, showing that our approach indeed allows to avoid having to choose a single signal for clustering ahead of time.

Table 4.6: ARI and CP for different methods

Method	*E. coli*		*Y. pseudotuberculosis*		*M. tuberculosis*	
	ARI	CP	ARI	CP	ARI	CP
Phydelity	0.76	0.93	0.23	0.94	-	0.92
TreeCluster	0.08	0.96	0.01	0.90	-	0.74
PathOGiST						
ILP: SNP	**0.92**	**1.0**	**0.96**	**0.98**	-	0.56
ILP: MLST	0.90	0.95	0.94	0.94	-	0.95
ILP: kWIP	0.90	**1.0**	**0.96**	0.94	-	0.57
ILP: CNV	-	-	-	-	-	**1.0**
ILP: SpoTyping	-	-	-	-	-	0.92
ILP: Consensus	0.91	0.85	**0.96**	0.97	-	0.57
C4: SNP	**0.92**	**1.0**	**0.96**	**0.98**	-	0.57
C4: MLST	0.90	0.95	0.94	0.94	-	0.95
C4: kWIP	0.90	0.99	**0.96**	0.94	-	0.60
C4: CNV	-	-	-	-	-	**1.0**
C4: SpoTyping	-	-	-	-	-	0.92
C4: Consensus	0.91	0.86	**0.96**	0.97	-	0.47

Chapter 5

Conclusion

In this work we described the open-source implementation of PathOGiST, an algorithmic framework for clustering bacterial isolates into epidemiologically related groups.

One of our key contributions is to introduce the paradigms of correlation clustering and consensus clustering to the bioinformatics community - while these have been commonly used in both data mining as well as theoretical computer science, they have not been used in bioinformatics so far, to the best of our knowledge. We expect that they will prove useful in other contexts and in other subfields of bioinformatics, given their simple formulation and the lack of a requirement for a pre-specified number of clusters.

Another contribution is to provide two implementations - one exact and one heuristic - of correlation clustering algorithms, and to tailor them to the problem at hand. More specifically, the column generation approach, whereby constraints that are violated by an initial solution are introduced in a subsequent iteration, rather than at the outset, appears to be the perfect fit for this problem when a lot of triangles (triples of isolates) are monochromatic (either all distant or all close to each other). In addition, it appears that the sign, rather than the magnitude, of the difference between a pairwise distance and the threshold is the main contributor, since even the C4 heuristic, which is based uniquely on the sign, tends to obtain high quality solutions. These observations are helpful in informing a strategy for future studies of the clustering problem.

In the future, we hope to address several challenges. First, an automated or semi-automated method for determining the optimal thresholds in a data-driven way would be necessary in order to extend the PathOGiST framework to the numerous pathogenic bacteria for which NGS data is available. Second, instead of a single output, a multiscale or hierarchical representation of the clusters may be helpful in order to provide the user with the flexibility of deciding on their own clustering granularity. Finally, some metadata, such as collection time or geographic location, may be fruitfully incorporated into the clustering approach in order to better inform the resolution of some groups of isolates. Despite these challenges, we believe that PathOGiST is a first step in the right direction, and we hope that it will provide an impetus for further exploration the problem of clustering bacterial isolates.

Bibliography

[1] Nader Alaridah, Erika Tång Hallbäck, Jeanette Tångrot, et al. Transmission dynamics study of tuberculosis isolates with whole genome sequencing in southern Sweden. *Scientific reports*, 9(1):4931, 2019.

[2] Metin Balaban, Niema Moshiri, Uyen Mai, et al. TreeCluster: clustering biological sequences using phylogenetic trees. *bioRxiv*, 2019.

[3] Nikhil Bansal, Avrim Blum, and Shuchi Chawla. Correlation clustering. *Mach. Learn.*, 56:89–113, 2004.

[4] Paola Bonizzoni, Gianluca Della Vedova, Riccardo Dondi, and Tao Jiang. On the approximation of correlation clustering and consensus clustering. *Journal of Computer and System Sciences*, 74:671–696, 2008.

[5] Josephine Bryant, Claire Chewapreecha, and Stephen D Bentley. Developing insights into the mechanisms of evolution of bacterial pathogens from whole-genome sequences. *Future Microbiology*, 7(11):1283–1296, 2012. PMID: 23075447.

[6] Lu Cheng, Thomas R Connor, Jukka Sirén, et al. Hierarchical and spatially explicit clustering of DNA sequences with BAPS software. *Mol. Biol. Evol.*, 30:1224–1228, 2013.

[7] Xiangyu Deng, Henk C. den Bakker, and Rene S. Hendriksen. Genomic epidemiology: Whole-genome-sequencing–powered surveillance and outbreak investigation of foodborne bacterial pathogens. *Annual Review of Food Science and Technology*, 7(1):353–374, 2016.

[8] William J Faison, Alexandre Rostovtsev, Eduardo Castro-Nallar, Keith A Crandall, Konstantin Chumakov, Vahan Simonyan, and Raja Mazumder. Whole genome single-nucleotide variation profile-based phylogenetic tree building methods for analysis of viral, bacterial and human genomes. *Genomics*, 104(1):1–7, 2014.

[9] Pedro Feijao, Hua-Ting Yao, Dan Fornika, et al. MentaLiST-a fast MLST caller for large MLST schemes. *Microb. Genom.*, 4, 2018.

[10] R. Fulkerson G. Dantzig and S. Johnson. Solution of a large-scale traveling salesman problem. *Operations Research*, 2:393–410, 1954.

[11] Olivier Gascuel. BIONJ: an improved version of the NJ algorithm based on a simple model of sequence data. *Molecular biology and evolution*, 14(7):685–695, 1997.

[12] Jennifer L Guthrie, Andy Delli Pizzi, David Roth, et al. Genotyping and whole-genome sequencing to identify tuberculosis transmission to pediatric patients in British Columbia, Canada, 2005–2014. *J. Infect. Dis.*, 40:1–9, 2018.

[13] Alvin X Han, Edyth Parker, Sebastian Maurer-Stroh, et al. Inferring putative transmission clusters with Phydelity. *bioRxiv*, 2019.

[14] William P Hanage, Christophe Fraser, and Brian G Spratt. Sequences, sequence clusters and bacterial species. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 361(1475):1917–1927, 2006.

[15] Hollie-Ann Hatherell, Caroline Colijn, Helen R Stagg, Charlotte Jackson, Joanne R Winter, and Ibrahim Abubakar. Interpreting whole genome sequencing for investigating tuberculosis transmission: a systematic review. *BMC medicine*, 14:21, mar 2016.

[16] Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of classification*, 2(1):193–218, 1985.

[17] Teemu Kallonen, Hayley J Brodrick, Simon R Harris, et al. Systematic longitudinal survey of invasive *Escherichia coli* in England demonstrates a stable population structure only transiently disturbed by the emergence of ST131. *Genome Res.*, 27:1437–1449, 2017.

[18] Mary Elizabeth Kaufmann. Pulsed-field gel electrophoresis. In *Molecular Bacteriology*, pages 33–50. Springer, 1998.

[19] Lynn Kennemann, Xavier Didelot, Toni Aebischer, Stefanie Kuhn, Bernd Drescher, Marcus Droege, Richard Reinhardt, Pelayo Correa, Thomas F. Meyer, Christine Josenhans, Daniel Falush, and Sebastian Suerbaum. Helicobacter pylori genome evolution during human infection. *Proceedings of the National Academy of Sciences*, 108(12):5033–5038, 2011.

[20] Sergei L Kosakovsky Pond, Steven Weaver, Andrew J Leigh Brown, and Joel O Wertheim. Hiv-trace (transmission cluster engine): a tool for large scale molecular epidemiology of hiv-1 and other rapidly evolving pathogens. *Molecular biology and evolution*, 35(7):1812–1819, 2018.

[21] Bruce R. Levin and Carl T. Bergstrom. Bacteria are different: Observations, interpretations, speculations, and opinions about the mechanisms of adaptive evolution in prokaryotes. *Proceedings of the National Academy of Sciences*, 97(13):6981–6985, 2000.

[22] Nicholas J Loman and Mark J Pallen. Twenty years of bacterial genome sequencing. *Nature Reviews Microbiology*, 13(12):787, 2015.

[23] Martin CJ Maiden, Jane A Bygraves, Edward Feil, et al. Multilocus sequence typing: a portable approach to the identification of clones within populations of pathogenic microorganisms. *Proceedings of the National Academy of Sciences*, 95(6):3140–3145, 1998.

[24] Martin CJ Maiden, Melissa J Jansen Van Rensburg, James E Bray, et al. MLST revisited: the gene-by-gene approach to bacterial genomics. *Nature Reviews Microbiology*, 11(10):728, 2013.

[25] Christopher Manning, Prabhakar Raghavan, and Hinrich Schütze. Introduction to information retrieval. *Natural Language Engineering*, 16(1):100–103, 2010.

[26] Mehrdad Mansouri, Julian Booth, Margaryta Vityaz, et al. PRINCE: accurate approximation of the copy number of tandem repeats. In *18th International Workshop on Algorithms in Bioinformatics (WABI 2018)*, pages 20:1–20:13, 2018.

[27] Conor J. Meehan, Pieter Moris, Thomas A. Kohl, et al. The relationship between transmission time and clustering methods in *Mycobacterium tuberculosis* epidemiology. *EBioMedicine*, 37:410–416, 2018.

[28] Matthias Merker, Thomas A. Kohl, Stefan Niemann, and Philip Supply. *The Evolution of Strain Typing in the Mycobacterium tuberculosis Complex*, pages 43–78. Springer International Publishing, Cham, 2017.

[29] Kevin D. Murray, Christfried Webers, Cheng Soon Ong, et al. kWIP: The k-mer weighted inner product, a de novo estimator of genetic similarity. *PLoS Comput. Biol.*, 13:1–17, 2017.

[30] Nam-Phuong Nguyen, Tandy Warnow, Mihai Pop, and Bryan White. A perspective on 16s rrna operational taxonomic unit clustering using sequence similarity. *NPJ biofilms and microbiomes*, 2:16004, 2016.

[31] Sophie Octavia, Qinning Wang, Mark M Tanaka, Sandeep Kaur, Vitali Sintchenko, and Ruiting Lan. Delineating community outbreaks of Salmonella enterica serovar Typhimurium by use of whole-genome sequencing: insights into genomic variability within an outbreak. *Journal of clinical microbiology*, 53(4):1063–1071, apr 2015.

[32] Brian D Ondov, Todd J Treangen, Páll Melsted, et al. Mash: fast genome and metagenome distance estimation using minhash. *Genome biology*, 17(1):132, 2016.

[33] Xinghao Pan, Dimitris S. Papailiopoulos, Samet Oymak, et al. Parallel correlation clustering on big graphs. In *Advances in Neural Information Processing Systems 28 (NIPS 2015)*, pages 82–90, 2015.

[34] M. Pérez-Losada, M. Arenas, and E. Castro-Nallar. Multilocus sequence typing of pathogens. In *Genetics and Evolution of Infectious Diseases*, pages 383–404. Elsevier, 2017.

[35] Pubmlst - public databases for molecular typing and microbial genome diversity. `https://pubmlst.org/`. Accessed: 2018-11-23.

[36] Manon Ragonnet-Cronin, Emma Hodcroft, Stéphane Hué, Esther Fearnhill, Valerie Delpech, Andrew J Leigh Brown, and Samantha Lycett. Automated analysis of phylogenetic clusters. *BMC bioinformatics*, 14(1):317, 2013.

[37] MB Reed, VK Pichler, F McIntosh, et al. Major Mycobacterium tuberculosis lineages associate with patient country of origin. *Journal of Clinical Microbiology*, 47:1119–1128, 2009.

[38] Andreas Roetzer, Roland Diel, Thomas A. Kohl, Christian Rückert, Ulrich Nübel, Jochen Blom, Thierry Wirth, Sebastian Jaenicke, Sieglinde Schuback, Sabine Rüsch-Gerdes, Philip Supply, Jörn Kalinowski, and Stefan Niemann. Whole genome sequencing versus traditional genotyping for investigation of a mycobacterium tuberculosis outbreak: A longitudinal molecular epidemiological study. *PLOS Medicine*, 10(2):1–12, 02 2013.

[39] Torsten Seemann. Snippy, 2015.

[40] Koichiro Tamura and Masatoshi Nei. Estimation of the number of nucleotide substitutions in the control region of mitochondrial dna in humans and chimpanzees. *Molecular biology and evolution*, 10(3):512–526, 1993.

[41] Gilles Vergnaud and Christine Pourcel. Multiple locus variable number of tandem repeats analysis. In *Molecular Epidemiology of Microorganisms*, pages 141–158. Springer, 2009.

[42] Luc Villandré, Aurélie Labbe, Bluma Brenner, Michel Roger, and David A Stephens. Dm-phyclus: a bayesian phylogenetic algorithm for infectious disease transmission cluster inference. *BMC bioinformatics*, 19(1):324, 2018.

[43] Irene Vrbik, David A Stephens, Michel Roger, and Bluma G Brenner. The gap procedure: for the identification of phylogenetic clusters in hiv-1 sequence data. *BMC bioinformatics*, 16(1):355, 2015.

[44] Timothy M Walker, Camilla LC Ip, Ruth H Harrell, Jason T Evans, Georgia Kapatai, Martin J Dedicoat, David W Eyre, Daniel J Wilson, Peter M Hawkey, Derrick W Crook, Julian Parkhill, David Harris, A Sarah Walker, Rory Bowden, Philip Monk, E Grace Smith, and Tim EA Peto. Whole-genome sequencing to delineate mycobacterium tuberculosis outbreaks: a retrospective observational study. *The Lancet Infectious Diseases*, 13(2):137 – 146, 2013.

[45] Deborah A Williamson, Sarah L Baines, Glen P Carter, et al. Genomic insights into a sustained national outbreak of *Yersinia pseudotuberculosis*. *Genome Biol. Evol.*, 8:3806–3814, 2017.

[46] E. Xia, Y.-Y. Teo, and R. T.-H. Ong. SpoTyping: fast and accurate in silico mycobacterium spoligotyping from sequence reads. *Genome Medicine*, 8(19), 2016.