ALGORITHMIC DATA FUSION METHODS FOR TUBERCULOSIS

By

Cagri Ozcaglar

A Thesis Submitted to the Graduate

Faculty of Rensselaer Polytechnic Institute

in Partial Fulfillment of the

Requirements for the Degree of

DOCTOR OF PHILOSOPHY

Major Subject: COMPUTER SCIENCE

Approved by the
Examining Committee:

____
Bülent Yener, Thesis Adviser

____
Kristin P. Bennett, Member

____
Mohammed Zaki, Member

____
Chris Bystroff, Member

____
Qiang Ji, Member

Rensselaer Polytechnic Institute
Troy, New York

July 2012
(For Graduation August 2012)

CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGMENT

First and foremost, I would like to thank my advisor Prof. Bülent Yener for his support and guidance throughout my professional development. His visionary ideas in data fusion methods via matrix and tensor decomposition made this thesis possible. I would also like to thank him for believing in me during my studies at RPI.

I would like to thank Prof. Kristin Bennett for giving me the opportunity to conduct research in TB-Insight project. She has shown me that all research problems can be reduced to an optimization problem, which keeps me motivated for my current and future research. I would like to express my gratitude to Prof. Mohammed Zaki for our continuing journey which started with his teachings in data mining in 2007. His stimulating suggestions in various aspects of my research will start to sparkle as you turn the next pages. I would also like to thank other members of my committee. Prof. Chris Bystroff gave me invaluable insights into phylogenetic analysis through his teachings in bioinformatics and through our discussions on my research. Prof. Qiang Ji directed me on alternative tensor decomposition methods and our fruitful discussions will help me in my future research. Finally, I would like to thank my M.S. advisor Prof. Sibel Adali and M.S. co-advisor Prof. Boleslaw Szymanski for their continued support.

I would like to thank my labmates who provided a supportive working environment. I would like to express my gratitude to Amina Shabbeer who has been a loyal constant co-author one way or another. I would like to thank Dr. Minoo Aminian for always being available to share ideas on our research. I would also like to thank my other labmates: Srivatsan Raghavan, Janani Ranganathan, Jacob Katz, Dan Croft, and Kane Hadley. Finally, thanks to my labmates during my master's studies, Antonio Penta and Alvaro Graves, for good old times at RPI.

Life in Troy exists only if you have great friends. I would like to thank Sahin Cem Geyik, Onur Tekdas, Alper Ayvaci, Akintayo Holder, Jing Fu, Ning Liu and Cem Onur Karatas who made my graduate years enjoyable and memorable. I would like to thank my former roommates Ali Civril and Kadir Elitok, and my final room-

mate Cagri Yanik for sharing great times. I would like to thank Hilmi Yildirim, Bugra Caskurlu, Ahmet Camtepe, Basak Oztan and Mohammad Al Hasan for sharing their wise suggestions. I would like to thank my friends in different locations of the world for continuously supporting me: Murat Ozyurek, Ismet Zeki Yalniz, Murat Tasci, Mustafa Oztekin. Finally, thanks to Isil for all her support.

Last but not least, I want to express my deepest gratitude to my family for their invaluable support and love. I dedicate this thesis to my father who convinced me to stay in academia, to my mother who has been available to listen at all times, and to my brother, Caglar, for his continuous company.

ABSTRACT

Exponentially-growing genomic data after the advent of gene sequencing technologies shifted the emphasis on to the analysis of many datasets from as many sources as possible. Data from multiple sources in the form of matrices and tensors can be analyzed separately, or they can be coupled and decomposed simultaneously. This data deluge is also observed in patient datasets of tuberculosis (TB), an infectious disease caused by *Mycobacterium tuberculosis* complex (MTBC). Epidemiologists, clinicians, and health care practitioners aim to find transmission routes, detect or rule out possible outbreaks, and control TB. For this purpose, patient isolates are routinely genotyped by multiple biomarkers which include spacer oligonucleotide types (spoligotypes) and Mycobacterial Interspersed Repetitive Units - Variable Number Tandem Repeats (MIRU-VNTR). Now it remains to make inferences from this data congestion. In this thesis, we propose algorithmic data fusion methods for tuberculosis using multiple sources of information from MTBC strains and TB patients.

In the first study, we propose the Tensor Clustering Framework (TCF) on multiple-biomarker tensors (MBT) and subdivide major lineages of MTBC into sublineages via genomic data fusion. The MBT holds data from two biomarkers, spoligotypes and MIRU patterns. We factorize the MBT into its component matrices using multiway models. Based on the component matrix of strain mode, we cluster MTBC strains into sublineages. Our new definition of sublineages based on two biomarkers confirms some of the existing sublineages, and suggests subdividing or merging other sublineages.

In the second study, we propose a new mutation model of spoligotypes based on both spoligotypes themselves and MIRU patterns. The model uses a maximum parsimony method based on three genetic distance measures on these two biomarkers. The resulting putative mutation history of spoligotypes depicted via a spoligoforest shows notable topological attributes. Number of descendant spoligotypes follows a power-law distribution. In addition, number of mutations at each spacer in the

DR region follows a spatially bimodal distribution. Based on this observation, we built two alternative models for mutation length frequency: Starting Point Model (SPM) and Longest Block Model (LBM). Both models plausibly fit mutation length frequency distribution in the spoligoforest.

In the third study, we propose the Unified Biclustering Framework (UBF) for host-pathogen association analysis of tuberculosis patients via genome-phenome data fusion. UBF is flexible in the sense that we can incorporate genetic distance between MTBC strains, spatial distance between TB patients, and time into domain knowledge, and factorize these joint datasets via coupled matrix-matrix and matrix-tensor factorization. We calculate feature pattern similarity matrix of (spoligotype, country) pairs and use it as input to our novel density-invariant biclustering algorithm. Finally, we select statistically significant biclusters using average best-match score. The resulting biclusters verify some of the well-known host-pathogen associations between MTBC strains and geographic distribution of their hosts, as well as suggest new patient-strain relationships.

CHAPTER 1
INTRODUCTION

Tuberculosis (TB) is one of the most fatal infectious diseases worldwide [1]. It is acquired through airborne infection and transmission. Among all TB infections, 90% of the cases remain latent, while in only 10% of the cases the patient is infected with active TB [2]. Most active TB cases involve infection in the lungs, which are referred as pulmonary TB cases. Tuberculosis infections on other sites of the body cause extrapulmonary TB [3]. *Mycobacterium tuberculosis* complex (MTBC) is the bacteria which causes TB. DNA fingerprinting methods are used to discriminate and identify genetically related MTBC strains. Genetic diversity of MTBC strains leads to different levels of pathogenicity, transmissivity, virulence and drug resistance. The genetic variation of MTBC also follows a pattern on geographic distribution of their hosts and their attributes [4]. Therefore, it is highly desirable to define the borders of genetic variability of MTBC strains, explain mutation mechanism of their genetic markers, and make inferences on host populations.

1.1 Tuberculosis

Tuberculosis is an infectious disease transmitted by its pathogen MTBC. In this section, we describe the symptoms and treatment options of TB, and statistics based on long-term outcomes of TB. Then, we describe the MTBC genome and present the biomarkers used for MTBC genotyping.

1.1.1 The disease

Tuberculosis is a bacterial disease affecting the lungs in most TB cases, leading to pulmonary TB. The symptoms include prolonged bad cough, chest pain, coughing up blood, fever, weight loss, fatigue and night sweat. According to the World Health Organization (WHO), one third of the human population is infected with latent or

* Portions of this chapter previously appeared as: C. Ozcaglar, A. Shabbeer, S. L. Vandenberg, B. Yener, and K. P. Bennett, Epidemiological models of *Mycobacterium tuberculosis* complex infections, Math. Biosciences, vol. 236, no. 2, pp. 77-96, 2012.

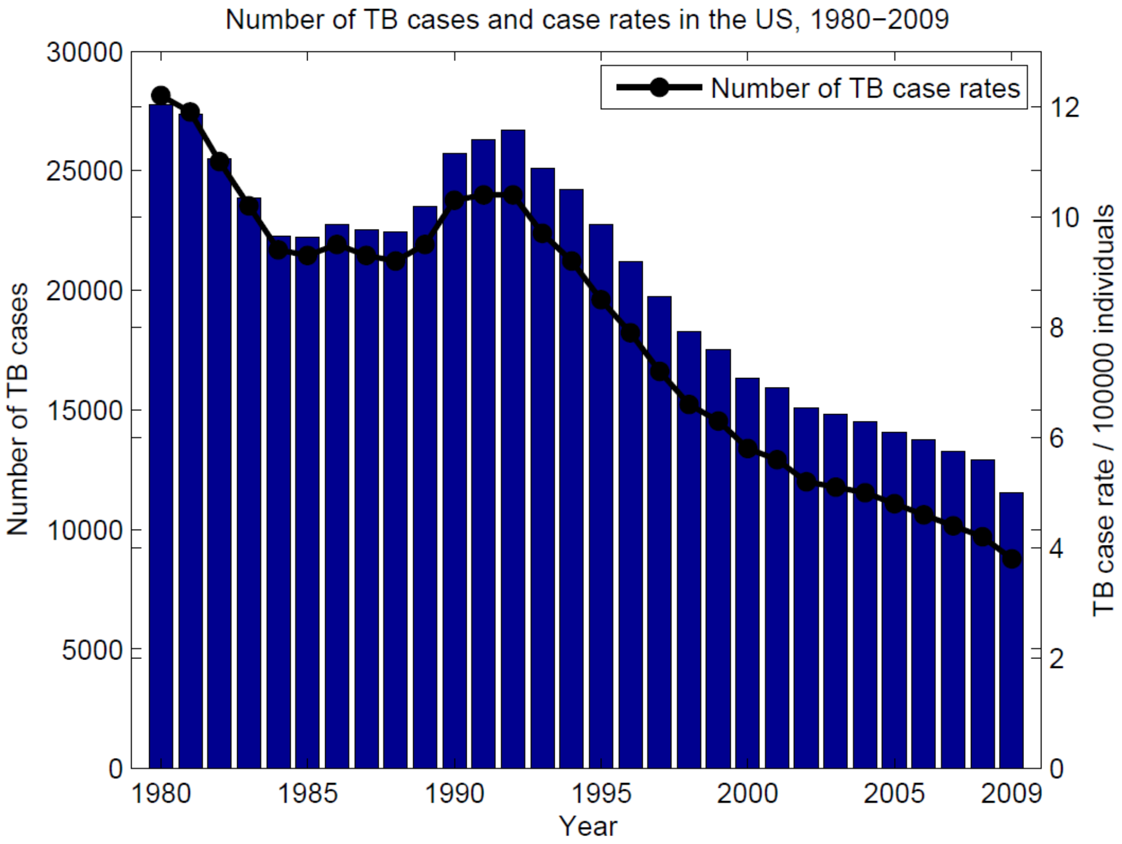


Figure 1.1: Number of cases and case rates per 100,000 individuals in the US between 1980-2009 shows a general downward trend with the exception of a sudden rise in 1990s. The plot is generated using the data from [5].

active TB [1]. Tuberculosis can be treated by anti-tuberculosis drugs which require taking pills for 6-9 months. TB infection can also be prevented by Bacillus Calmette Guérin (BCG) vaccine before infection.

Tuberculosis case counts and case rates have changed in the US and worldwide over the years. Figure 1.1 shows the number of TB cases and case rates in the US from 1980 to 2009. The number of cases and case rates both follow a decreasing trend, with the exception of increasing TB cases and case rates in the early 1990s. The increase in this period was attributed to several factors: the HIV epidemic in the early 1990s leading to HIV/TB co-infection, the emergence of drug resistant TB, immigration to the US from developing countries, and increased mass transportation [6–8]. Although the numbers look optimistic, detection and treatment rates of tuberculosis in developing countries are lower, which causes more than two million people to die from TB every year.

Tuberculosis has long latency periods and progresses slowly inside the individual's body. At the population level, this slow progression results in tuberculosis epidemics that span long time intervals. Therefore, epidemiological models are needed to estimate the long-term effects of TB epidemics [9]. One such TB epidemic took place in Europe at the beginning of 17th century, and continued for the next 200 years, which is also known as the *White Plague*.

1.1.2 The pathogen

Mycobacterium tuberculosis complex (MTBC) is the airborne pathogen of tuberculosis. The MTBC is comprised of the species *M. tuberculosis*, *M. africanum*, *M. bovis*, *M. canettii*, *M. microti*, and *M. pinnipedii*. The ideal approach to access genetic variation among MTBC strains is to sequence and compare their whole genomes [10]. Cole et al. sequenced the complete genome of *M. tuberculosis* H37Rv strain which possesses 4,411,529-bp and made a huge impact in TB research in 1998 [11]. Soon after, in 2003, Garnier et al. presented the 4,345,492-bp genome sequence of *M. bovis* AF21222/97, and compared it to the complete genome sequence of *M. tuberculosis* H37Rv [12]. However, whole genome sequencing is time-consuming, labor-intensive, and slow for TB control and prevention. Therefore, only the genomic loci with enough dissimilarity among strains are used to genotype MTBC. DNA fingerprinting of MTBC strains reveals differences among MTBC isolates which are genotyped by multiple biomarkers. These biomarkers are spacer oligonucleotide types (spoligotypes), Mycobacterial Interspersed Repetitive Units - Variable Number Tandem Repeats (MIRU-VNTR), IS*6110* restriction fragment length polymorphisms (RFLP), long sequence polymorphisms (LSPs), and single nucleotide polymorphisms (SNPs) [13, 14]. MTBC strains can be genotyped with these biomarkers in order to detect or rule out outbreaks, identify and distinguish MTBC strains into distinct lineages, and track transmission routes of TB.

The most commonly used MTBC genotyping methods are spoligotyping, MIRU typing and RFLP analysis [10, 16]. Spoligotyping is based on the polymorphisms in

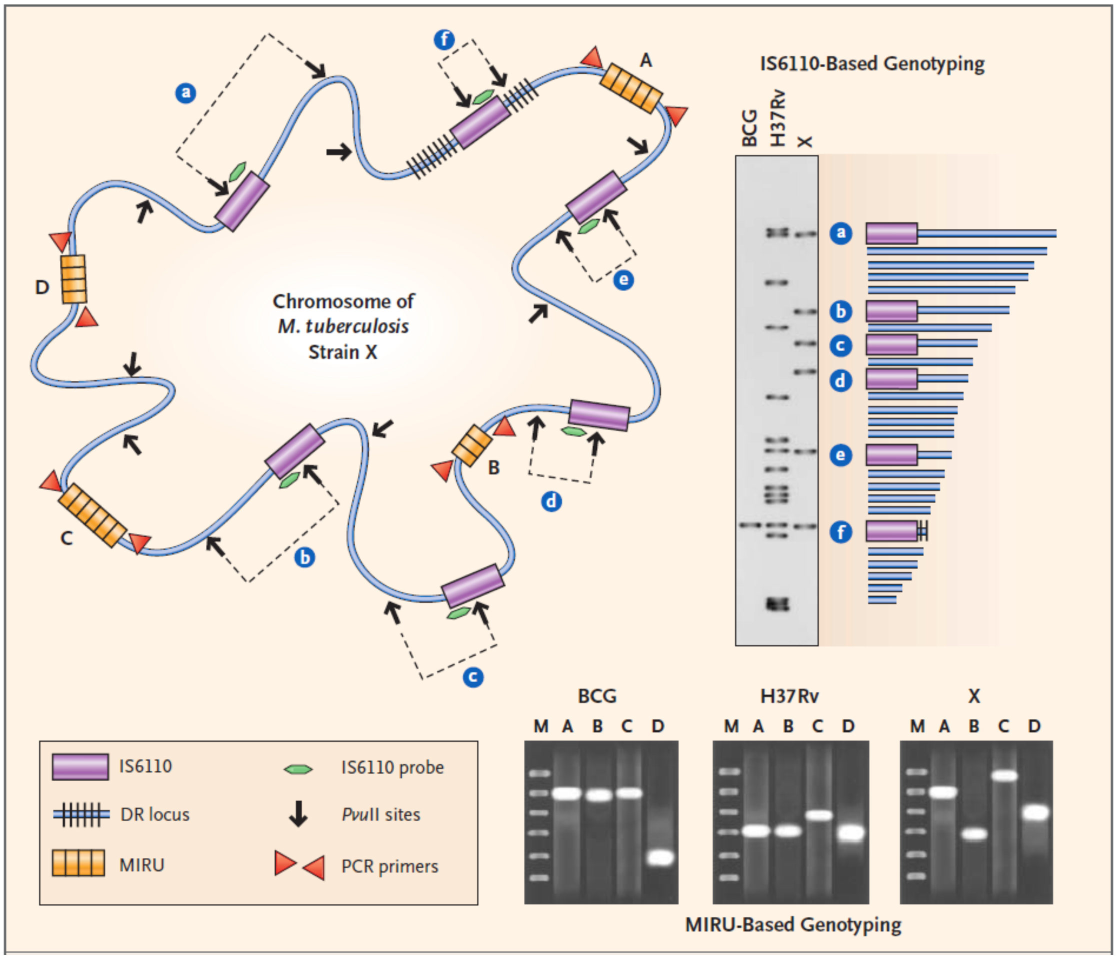


Figure 1.2: MTBC genome. Spoligotyping is based on the polymorphisms in the DR locus. MIRU typing is based on the polymorphisms in MIRU loci. RFLP analysis is based on the copy number of IS*6110* insertion sequences. The figure was taken from a study by Barnes et al. [15].

the direct repeat (DR) region, which consists of direct repeats separated by spacer sequences [17]. The method uses 43 spacers, which is represented as a 43-bit binary sequence, where zeros and ones represent absence and presence of spacers respectively. MIRU-VNTR analysis is based on the polymorphisms in 41 mini-satellite region dispersed within the intergenic regions of MTBC [18]. Out of 41 MIRU loci, 12, 15, and 24 loci MIRU pattern analysis formats are used. MIRU patterns are represented as n-bit digit sequence, where n is the number of loci used in the MIRU analysis. RFLP analysis is the gold standard for MTBC genotyping. Strains are typed based on the copy number and the variability in the positions of IS*6110* insertion sequences [14]. RFLP analysis has higher discriminatory power compared to spoligotyping and MIRU-VNTR typing. Figure 1.2 from a study by Barnes et al. shows the physical distribution of DR locus, MIRU loci, and IS*6110* insertion sequences on MTBC genome [15].

1.2 Motivation

Availability of multiple biomarkers of MTBC strains and patient attributes enhances the efforts for TB control and prevention. Classification of MTBC strains using multiple biomarkers leads to new lineages. Similarly, mutation mechanism of one biomarker can be more accurately described via an additional independent biomarker. Finally, genetically similar MTBC strains can infect phenotypically similar host subpopulations with high association and possible adaptation. In this section, we briefly detail the motivation behind the use of data from multiple sources in these three problems.

Groups of genetically close MTBC strains are interchangeably called sub/families, sub/clades and sub/lineages [19–26]. Throughout this thesis, we will refer to them as lineages and sublineages. In earlier TB databases such as SpolDB3 and SpolDB4, only one biomarker of MTBC was used to define genetic lineages and sublineages [19, 27]. However, it is advantageous to use multiple biomarkers and increase discriminatory power of biomarkers for strain classification [28]. Recently developed models such as Conformal Bayesian Network (CBN) and Knowledge-based Bayesian Network (KBBN) made use of multiple biomarkers for classification of MTBC strains into lineages and sublineages respectively [20–22]. Similarly, multiple biomarkers can be used to enrich or correct strain differentiation and classification via genomic data fusion. Therefore, with the advent of MTBC genotyping, new data fusion methods are needed to combine multiple biomarker data in one framework.

Evolutionary analysis of MTBC depends on mutation mechanism of each individual biomarker. Since DR locus, MIRU loci and IS*6110* insertion sequences are distributed randomly on MTBC genome, it is assumed that they evolve independently, with some rare exceptions on the dependence of IS*6110* transposition and

DR evolution [29]. To refine evolutionary history of a biomarker on MTBC genome, it is beneficial to use an independent biomarker and incorporate mutation mechanisms of multiple biomarkers in one evolutionary scenario. For instance, Fenner et al. revisited DR evolution and found evidence of rare convergent evolution in the DR region using SNPs, 24-loci MIRU-VNTR and genomic deletions [30]. This suggests that use of multiple biomarkers in one evolutionary scenario can expose unknown or unforeseen mutation mechanism of biomarkers, which is hard to observe by examining the evolution of one biomarker at a time. Therefore, using mutation mechanism of multiple biomarkers in one evolutionary scenario via phylogenetic analysis can lead to new insights about the evolution of individual biomarkers.

Stable and variable host-pathogen associations between MTBC strains and TB patients have been observed in earlier studies [4, 31]. This suggests that joint modeling of MTBC genotype and patient phenotype data can identify new lineages and sublineages, most of which are named according to their dominance in particular geographical regions, leading to phylogeographic lineage names. In addition, genetic proximity between MTBC strains can be incorporated into association analysis to enforce most likely mutation events to occur. Similarly, spatial proximity between TB patients can be incorporated into analysis to favor most likely transmission events. Therefore, genome-phenome data fusion methods incorporating relations among MTBC strain genome and relations among TB patient phenome can enhance host-pathogen association analysis of tuberculosis patients. In this thesis, we propose algorithmic data fusion methods for these three problems.

1.3 Our contributions

In this thesis, we propose new algorithmic data fusion methods for MTBC strains and TB patients. Our contributions are threefold. First, we subdivide major lineages of MTBC into sublineages based on two biomarkers using the *Tensor Clustering Framework (TCF)* on *Multiple-biomarker tensors (MBT)*. Second, we present a new evolution model for spoligotypes based on two biomarkers, and analyze topological attributes of the biological network reflecting the mutation history. Third, we propose the *Unified Biclustering Framework (UBF)*, and find biclusters which map to existing host-pathogen associations as well as new ones. A summary of our contributions is as follows:

- **Sublineage structure of MTBC:** We propose the *Tensor Clustering Framework (TCF)* on *Multiple-biomarker tensors (MBT)* and subdivide major lineages of MTBC into sublineages via genomic data fusion [23–25]. The multiple-biomarker tensor holds information about two biomarkers for each MTBC strain. We then apply multiway models on the multiple-biomarker tensor and decompose it into component matrices. Based on the component matrix of the strain mode, we cluster MTBC strains into groups of genetically similar MTBC strains. We compare these tensor sublineages to an existing sublineage definition based only on spoligotypes. We confirm some of the existing sublineages and suggest subdivision or merging of other sublineages.

- **Mutation of spoligotypes:** We propose a mutation model of spoligotypes based on both spoligotypes and MIRU patterns of MTBC strains [32,33]. The model is based on a maximum parsimony method using three genetic distance measures defined on these two biomarkers. The resulting putative mutation history of spoligotypes depicted via a spoliogoforest reveals a power-law distribution on the number of descendant spoligotypes, and spatially bimodal distribution of number of mutations at each spacer in the DR region. Based on this distribution, we built two alternative models for mutation length frequency: Starting Point Model (SPM) and Longest Block Model (LBM). Both models accurately describe the mutation length frequency distribution of spoligotypes.

- **Host-pathogen association:** We propose the *Unified Biclustering Framework (UBF)* for host-pathogen association analysis via genome-phenome data fusion [34]. UBF is flexible in the sense that genetic proximity between MTBC strains, spatial proximity between TB patients, and time can be incorporated into association analysis. This refines our search by favoring most likely mutation and transmission events. The biclustering results show high correlation between spoligotypes of MTBC strains and their hosts from particular countries. These biclusters point to some of the existing patient-strain relationships, and reveal new associations.

1.4 Organization

The organization of this thesis is as follows: In Chapter 2, we give an overview of post-genomic data analysis. In Chapter 3, we present the sublineage structure of MTBC using the Tensor Clustering Framework on multiple-biomarker tensors. In Chapter 4, we present the mutation model of spoligotypes based on multiple biomarkers of MTBC. In Chapter 5, we present the Unified Biclustering Framework and propose existing and new host-pathogen associations. In Chapter 6, we end with conclusions and future directions for our research.

CHAPTER 2
BACKGROUND: POST-GENOMIC DATA ANALYSIS

Following the completion of Human Genome Project with the release of the human genome in 2001, pre-genomic era ended and yielded to post-genomic era [35]. With the advent of automated fast gene sequencing techniques, available genomic data is growing faster than ever. New methods are built to extract knowledge from raw genomic data for interpretation of genome from biological or evolutionary perspectives [36]. Many more hypotheses, including gene-gene interactions, gene-environment interactions, uneven contributions of multiple genes to a disease, and genome-phenome relationships remain unexamined and require more complex methods. In this section, we briefly review data mining and machine learning methods for the analysis of exponentially-growing post-genomic data. These methods include classification, clustering, biclustering, multiway modeling, and phylogenetic analysis of genomic data.

2.1 Classification and Clustering

Classification and clustering are commonly used for genomic data. Classification assigns labels to data points via a model, and it is a supervised learning method. Clustering methods on the other hand group data points into close and compact subgroups based on the structure of the data, and it is an unsupervised learning method. Next, we describe classification, clustering, and present tools for both tasks on MTBC strains.

2.1.1 Classification

Classification is the task of finding a model that distinguishes data classes in order to predict predefined classes of unlabeled data points [37]. The derived model is learned based on a set of training data with labeled data points, and results in a function $\mathbf{f}$ that maps each attribute set $\mathbf{x}$ to one of the predefined classes $\mathbf{y}$ [38]. Because the class of each data point in the training set is known, classification is a supervised learning task. Examples of biological data classification methods include classification of yeast genes into functional categories using support vector machines (SVM) and classification of hereditary breast cancer data via partial least squares (PLS) [39, 40].

2.1.2 Clustering

Clustering is the task of grouping a dataset into classes of similar data points [37]. The goal is to maximize intra-cluster similarity and minimize inter-cluster similarity. The classes of data points are unknown *a priori*, therefore clustering is an unsupervised learning task. Various clustering methods applied to post-genomic data include partition clustering algorithms such as k-means, hierarchical clustering, density-based clustering, and spectral clustering [41]. Examples of gene clustering include hierarchical clustering of yeast microarray data and human microarray data which identified genes with known similar functions [42].

2.1.3 Classification and Clustering tools for MTBC strains

Various methods and tools are developed to classify or cluster MTBC strains. A decision tree based method by Ferdinand et al. classified MTBC strains into 8 spoligotype-defined families and found that MIRU24 is an informative loci for classification [43]. The conformal Bayesian network (CBN) by Aminian et al. classified MTBC strains into 6 major lineages using spoligotypes and MIRU patterns, and confirmed that the classification accuracy is higher when both biomarkers are used [20]. The rule-based TB-Lineage model by Shabbeer et al. classified MTBC strains into 6 major lineages using spoligotypes and MIRU24 [26]. The knowledge-based Bayesian network (KBBN) by Aminian et al. classified MTBC strains into 45 sublineages based on expert-defined rules for spoligotypes and MIRU patterns [21].

There also exist tools for clustering MTBC strains into groups of similar strains. First came SPOTCLUST, an unsupervised probabilistic model based on spoligotypes, which confirmed existing lineages and suggested new lineages [44]. Affinity propagation based on deletions distance between spoligotypes by Borile et al. supported previously identified sublineages, identified new sublineages, and re-assigned MTBC strains in ill-defined sublineages [45].

2.2 Biclustering

Biclustering, also known as co-clustering, is a subset of clustering methods which allow simultaneous clustering of rows and columns of a matrix [46]. The concept was first introduced by Hartigan et al. in 1972 with the name *direct clustering* [47]. Mirkin et al. used the term biclustering in 1996, referring to the same method [46]. With the advances in gene sequencing technology and availability of inexpensive microarray experiments, biclustering became popular in microarray data analysis at the beginning of 21st century.

Biclustering simply refers to partitioning a matrix into coherent groups of submatrices. Figure 2.1 shows a submatrix of the data matrix which associates the corresponding rows and columns of the data, thereby forming a bicluster. The biclusters can overlap, and the union of biclusters does not have to cover the original matrix, as opposed to clustering. The definition of coherence in a submatrix depends on the type of bicluster. Biclusters can belong to one of these five major classes: biclusters with constant values, biclusters with constant values on rows, biclusters with constant values on columns, biclusters with coherent values, and biclusters with coherent evolutions [48]. Biclustering algorithms are designed to find one or more types of biclusters under investigation. Next, we present an overview of existing biclustering algorithms.

2.2.1 Algorithms

Various biclustering algorithms have been proposed using different heuristic approaches such as iterative row and column clustering combination, divide and conquer, greedy iterative search, exhaustive bicluster enumeration, and distribution parameter identification [48]. Direct clustering by Hartigan et al. treats a data matrix as one block, and uses a top-down divide and conquer row and column clustering [47]. At each iteration, the algorithm finds the row and column with highest within-block variance, and iterates until a predefined number of biclusters are found. Cheng and Church proposed several row/column removal/addition algorithms via a greedy heuristic to find δ-biclusters with highest mean squared residue [49]. Flexible Overlapped Biclustering (FLOC) algorithm by Yang et al. builds on the algorithms

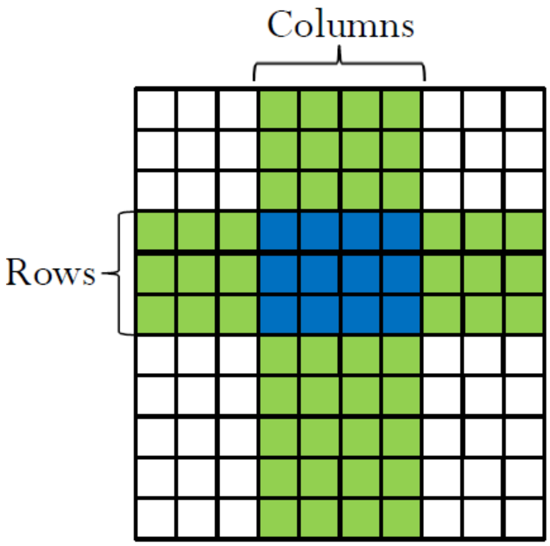


Figure 2.1: A bicluster refers to a submatrix within the data matrix. In the figure, the blue submatrix represents a bicluster which associates the corresponding rows and columns.

of Cheng and Church, and refines biclustering results after allowing biclusters to overlap [50]. Similarly, the Plaid model by Lazzeroni et al. finds a possibly overlapping bicluster at each iteration by minimizing a merit function [51].

Coupled Two-Way Clustering (CTWC) algorithm by Getz et al. finds a biclustering by iteratively applying one-way clustering algorithm to rows and columns of a matrix alternately [52]. Interrelated Two-Way Clustering (ITWC) algorithm by Tang et al. iteratively clusters rows and columns, combines clustering results of both dimensions, finds heterogeneous groups and finally reduces genes [53]. Spectral bipartitioning algorithm by Dhillon et al. uses the second left and right singular vectors of scaled data matrix to find biclusters [54]. Kluger et al. proposed a similar spectral biclustering algorithm which allows different numbers of clusters in both dimensions [55]. Sheng et al. proposed another biclustering method based on modeling rows and columns using independent multinomial distributions and estimating their parameters via Gibbs sampling [56]. The Order-Preserving Submatrix (OPSM) algorithm by Ben-Dor et al. finds submatrices such that there exists a permutation of columns under which the values in each row are strictly increasing [57]. Statistical-Algorithmic Method for Bicluster Analysis (SAMBA) by Tanay et al. uses exhaustive enumeration of biclusters and defines the ones corresponding to subgraphs with maximum weight as statistically significant biclusters [58]. Murali et al. proposed the xMOTIF algorithm which finds conserved gene expression motifs with maximum number of conserved genes [59]. Binary Inclusion-Maximal (BiMax) biclustering algorithm by Prelic et al. finds the maximal biclusters on a binary data matrix by searching for submatrices of all ones [60]. Density-constrained biclustering algorithm by Dao et al. finds biclusters such that the corresponding bipartite graph and at least one of its one-vertex-induced subgraphs have density above a threshold [61]. Next, we move on to applications of biclustering algorithms on genomic data.

2.2.2 Applications to genomic data

Biclustering methods have been widely used after the advances in gene sequencing and microarray technologies. Gene expression in microarray data links genotype and phenotype of genes or cells, which is critical to understanding biological processes such as gene regulation, gene function, gene evolution and the role of genes in diseases [62]. Therefore, biclustering applications on gene expression data are used to associate genes with specific conditions according to their expression level. Analysis of gene expression datasets yielded results in gene functional annotation, gene coregulation identification, and sample classification.

Most of the gene expression datasets used in biclustering applications belong to yeast or human cells. Tanay et al. used SAMBA algorithm on yeast transcriptional network and functional network [58, 63]. They identified the global organization of yeast system using associations of different modules to the functional network, and assigned functional annotation to uncharacterized yeast genes. Getz et al. used CTWC algorithm to analyze Leukemia samples [52]. They found a connection between T-cell-related genes and the subclassification of the acute lymphoblastic leukemia (ALL) samples into T cell and B cell ALL. They also found a stable partition of acute myeloid leukemia (AML) patients into groups of treated and untreated patients using a bicluster of cell-growth-related genes. Dao et al. used density-constrained biclustering algorithm on colon cancer and breast cancer data, and identified two dysregulated genes involved in TP53 signaling: GSE8671 for colon cancer, GSE3494 for breast cancer [61]. Colak et al. used densely connected biclustering (DECOB) algorithm on PPI/GI network of yeast and human gene expression data [64]. They showed that GO-term specific clusters of modules predict functional relationships more accurately. In all of these applications of biclustering algorithms, some biclusters have clear biological interpretation, and other biclusters are potential associations to be analyzed leading to directions in future research.

2.3 Multiway modeling

Multiway modeling is the extension of two-way data modeling to higher-order datasets [65, 66]. Multiway arrays, also referred as tensors, are higher-order generalizations of vectors and matrices. Tensors have a standardized terminology which is different than that of arrays of order 1 and 2 [67]. In the next section, we briefly review the preliminary concepts and notations of multiway modeling, present multiway models, algorithms for fitting multiway models, and applications of multiway modeling to genomic data.

2.3.1 Preliminaries and notation

Tensors are multiway arrays denoted as $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$, where the order of $\underline{\mathbf{X}}$ is $N > 2$. Each entry of $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ is represented by $x_{i_1 i_2 \ldots i_N}$. Each dimension of a tensor is called a *mode*. For example, the tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ has N modes. Fixing all but two indices of a tensor returns two-dimensional sections of the tensor, also called a *slice*. For the special case of three-dimensional tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, fixing the index in the first, second and third mode returns a *horizontal slice*, a *lateral slice*, and a *frontal slice* respectively. Fixing all but one index of a tensor returns vectors of the tensor, also called a *fiber*. For the 3-way tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, releasing the first, second, and third mode returns a *vertical fiber* or *column*, a *horizontal fiber* or *row*, and a *depth fiber* or *tube* respectively [67].

2.3.1.1 Vector and matrix products

Several vector and matrix products are frequently used in multiway analysis. An N-way rank-one tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ can be written as the outer product of

N vectors $\mathbf{a}_1 \in \mathbb{R}^{I_1}, \mathbf{a}_2 \in \mathbb{R}^{I_2}, \ldots, \mathbf{a}_N \in \mathbb{R}^{I_N}$ as in Equation (2.1):

$$\underline{\mathbf{X}} = \mathbf{a}_1 \circ \mathbf{a}_2 \circ \ldots \circ \mathbf{a}_N \tag{2.1}$$

such that

$$x_{i_1 i_2 \ldots i_N} = \mathbf{a}_{1 i_1}\, \mathbf{a}_{2 i_2}\, \ldots\, \mathbf{a}_{N i_N}$$

where the symbol $\circ$ represents *vector outer product.*

Kronecker product of two matrices $\mathbf{A} \in \mathbb{R}^{M \times N}$ and $\mathbf{B} \in \mathbb{R}^{P \times Q}$ is denoted as $\mathbf{A} \otimes \mathbf{B} \in \mathbb{R}^{MP \times NQ}$, and it is defined as in Equation (2.2).

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{bmatrix} \tag{2.2}$$

Khatri-Rao product is equivalent to column-wise Kronecker product. Khatri-Rao product of two matrices $\mathbf{A} \in \mathbb{R}^{I \times K}$ and $\mathbf{B} \in \mathbb{R}^{J \times K}$ is denoted as $\mathbf{A} \odot \mathbf{B} \in \mathbb{R}^{IJ \times K}$, and it is defined using Kronecker product of their columns as in Equation (2.3).

$$\mathbf{A} \odot \mathbf{B} = [\mathbf{a}_1 \otimes \mathbf{b}_1 \quad \mathbf{a}_2 \otimes \mathbf{b}_2 \quad \ldots \quad \mathbf{a}_K \otimes \mathbf{b}_K] \tag{2.3}$$

Hadamard product is the element-wise matrix product. The Hadamard product of two matrices $\mathbf{A} \in \mathbb{R}^{I \times J}$ and $\mathbf{B} \in \mathbb{R}^{I \times J}$ of the same size is denoted as in Equation (2.4).

$$\mathbf{A} * \mathbf{B} = \begin{bmatrix} a_{11}b_{11} & \cdots & a_{1J}b_{1J} \\ \vdots & \ddots & \vdots \\ a_{I1}b_{I1} & \cdots & a_{IJ}b_{IJ} \end{bmatrix} \tag{2.4}$$

2.3.1.2 Tensor products

Notation for tensor multiplication in multilinear algebra is different from matrix multiplication in linear algebra. The *n-mode vector product* of a tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ with a vector $\mathbf{v} \in \mathbb{R}^{I_n}$ is denoted by $\underline{\mathbf{X}} \times_n \mathbf{v} \in \mathbb{R}^{I_1 \times \ldots \times I_{n-1} \times I_{n+1} \times \ldots \times I_N}$. Each element of $\underline{\mathbf{X}} \times_n \mathbf{v}$ is calculated as follows:

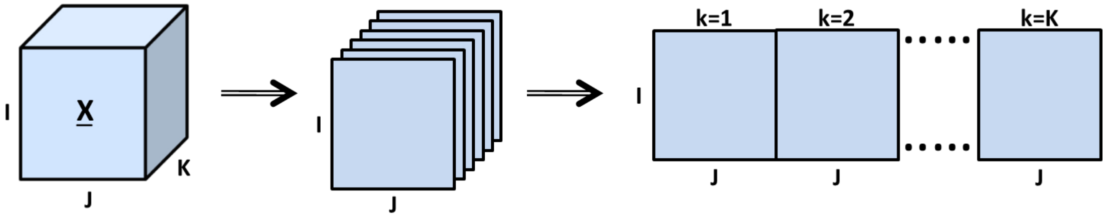


Figure 2.2: Matricization of 3-way tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$ in the first mode. The resulting matrix is $\mathbf{X}_{(1)} \in \mathbb{R}^{I \times JK}$.

$$(\underline{\mathbf{X}} \times_n \mathbf{v})_{i_1 \times \ldots \times i_{n-1} \times i_{n+1} \times \ldots \times i_N} = \sum_{i_n=1}^{I_n} x_{i_1 \times i_2 \times \ldots \times i_N} v_{i_n} \,. \tag{2.5}$$

The *n-mode matrix product* of a tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ with a matrix $\mathbf{A} \in \mathbb{R}^{J \times I_n}$ is denoted by is denoted by $\underline{\mathbf{X}} \times_n \mathbf{A} \in \mathbb{R}^{I_1 \times \ldots \times I_{n-1} \times J \times I_{n+1} \times \ldots \times I_N}$. Each element of $\underline{\mathbf{X}} \times_n \mathbf{A}$ is calculated as follows:

$$(\underline{\mathbf{X}} \times_n \mathbf{A})_{i_1 \times \ldots i_{n-1} \times j \times i_{n+1} \times \ldots \times i_N} = \sum_{i_n=1}^{I_n} x_{i_1 \times i_2 \times \ldots \times i_N} a_{j i_n} \,. \tag{2.6}$$

2.3.1.3 Matricization

Tensors can be transformed into a matrix via a process called *matricization*, also known as *unfolding*. The mode-n matricization of tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ rearranges mode-n slices of the tensor as the columns of the resulting matrix, which is denoted as $\mathbf{X}_{(n)}$. Figure 2.2 shows the matricization of 3-way tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$ in the first mode. The resulting matrix is denoted as $\mathbf{X}_{(1)}$.

2.3.2 Multiway models

Multiway models are developed for decomposition of tensors into factor matrices. The two most commonly used tensor decomposition models are PARAFAC and Tucker3. In this section, we give a brief background on these multiway models.

2.3.2.1 PARAFAC

PARAFAC model is an extension of singular value decomposition to multilinear decomposition [65]. The model is simultaneously found by Carroll and Chang in the form of canonical decomposition (CANDECOMP), and by Harshman in the form of parallel factors (PARAFAC) [68, 69]. Therefore, the model is called CANDECOMP/PARAFAC decomposition, abbreviated as CP. In this study, we refer to this model as the PARAFAC model. The PARAFAC model can be represented as a linear combination of rank-1 tensors. An R-component PARAFAC model on 3-way tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$ is formulated as in Equation (2.7):

$$\underline{\mathbf{X}} \approx \sum_{r=1}^{R} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r = [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!] \tag{2.7}$$

where $\mathbf{a}_r \in \mathbb{R}^I$, $\mathbf{b}_r \in \mathbb{R}^J$ and $\mathbf{c}_r \in \mathbb{R}^K$ are columns of factor matrices $\mathbf{A} \in \mathbb{R}^{I \times R}$, $\mathbf{B} \in \mathbb{R}^{J \times R}$ and $\mathbf{C} \in \mathbb{R}^{K \times R}$ respectively. $[\![\mathbf{A},\ \mathbf{B},\ \mathbf{C}]\!]$ is the short-hand notation for the decomposition using Kruskal operator [70]. This model can also be represented in matrix notation using unfolded tensor as in Equation (2.8):

$$\mathbf{X}_{(1)} = \mathbf{A}\left(\mathbf{C} \odot \mathbf{B}\right)' + \mathbf{E}_{(1)} \tag{2.8}$$

where $\underline{\mathbf{E}} \in \mathbb{R}^{I \times J \times K}$ is the residual of the tensor decomposition. In the PARAFAC model, the number of components in each mode is the same, and the model is unique.

2.3.2.2 Tucker3

Tucker3 model is an extension of singular value decomposition to multilinear decomposition without the equality constraint on the number of components at each mode. The model is proposed by Tucker in 1963 and in his refined article in 1966 [71, 72]. Among the Tucker-family multiway models, the 3 in Tucker3 model indicates that the model returns the components of all modes, in particular, it returns components for all 3 modes of a 3-way array [73]. A (P, Q, R)-component Tucker3 model of 3-way array $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$ is formulated as in Equation (2.9):

$$\underline{\mathbf{X}} \approx \sum_{p=1}^{P} \sum_{q=1}^{Q} \sum_{r=1}^{R} g_{pqr}\, \mathbf{a}_p \circ \mathbf{b}_q \circ \mathbf{c}_r = [\![\underline{\mathbf{G}}; \mathbf{A}, \mathbf{B}, \mathbf{C}]\!] \tag{2.9}$$

where $\mathbf{a}_p \in \mathbb{R}^I$, $\mathbf{b}_q \in \mathbb{R}^J$ and $\mathbf{c}_r \in \mathbb{R}^K$ are columns of column-wise orthogonal factor matrices $\mathbf{A} \in \mathbb{R}^{I \times P}$, $\mathbf{B} \in \mathbb{R}^{J \times Q}$ and $\mathbf{C} \in \mathbb{R}^{K \times R}$ respectively. The tensor $\underline{\mathbf{G}} \in \mathbb{R}^{P \times Q \times R}$ is the core tensor and its entries show the level of interaction between different components [74]. $[\![\underline{\mathbf{G}};\, \mathbf{A},\, \mathbf{B},\, \mathbf{C}]\!]$ is the short-hand notation for the Tucker3 decomposition using the Tucker operator [75]. Tucker3 model can also be represented in matrix notation using unfolded tensor as in Equation (2.10):

$$\mathbf{X}_{(1)} = \mathbf{A}\mathbf{G}_{(1)}\left(\mathbf{C} \otimes \mathbf{B}\right)' + \mathbf{E}_{(1)} \tag{2.10}$$

where $\underline{\mathbf{E}} \in \mathbb{R}^{I \times J \times K}$ is the residual term. PARAFAC model is a special case of the Tucker3 model in which the core tensor is a cubical superdiagonal tensor such that $\underline{\mathbf{G}} \in \mathbb{R}^{R \times R \times R}$ with $g_{rrr} \neq 0$ [76]. Tucker3 model is more flexible than PARAFAC in that it allows different number of components at each mode, which in turn comes with a cost: Tucker3 decomposition of a tensor is not unique due to rotational freedom.

2.3.3 Algorithms

Given fixed number of components, there exist many algorithms for fitting multiway models to tensors. The most commonly used algorithms for tensor decomposition are based on alternating least squares (ALS) method. In this section, we present PARAFAC-ALS and Tucker3-ALS algorithms. We also briefly mention other algorithms for fitting multiway models.

2.3.3.1 PARAFAC-ALS

Based on the PARAFAC model formulation in Equation (2.8), the objective function is as follows:

$$L_1 = ||\mathbf{X}_{(1)} - \mathbf{A}\left(\mathbf{C} \odot \mathbf{B}\right)'||_F^2 \,. \tag{2.11}$$

To minimize L_1 in an alternating fashion, we minimize for one of the factor matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}$ while fixing others. Algorithm 1 describes the `PARAFAC-ALS` procedure for fitting an R-component PARAFAC model to $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$. PARAFAC-ALS can be initialized using a rational start such as generalized rank annihilation method (GRAM), a semi-rational start such as higher-order SVD, or a random start [77].

A recommended approach is to use both rational and semi-rational starting points as well as several random starting points in order to detect local minima if one exists. Convergence criterion of PARAFAC-ALS can be a combination of limiting the number of iterations and insufficient change in loss value, among many others. Other algorithms for fitting a PARAFAC model include alternating algorithms such as alternating slice-wise diagonalization and self-weighted alternating trilinear decomposition, derivative-based algorithms such as positive matrix factorization for 3-way arrays, damped Gauss Newton and CP-OPT, direct non-iterative algorithms such as GRAM and direct trilinear decomposition [78, 79].

Algorithm 1 `PARAFAC-ALS`($\underline{\mathbf{X}} \in \mathbb{R}^{I\times J\times K}$,$R$)

```
Initialize $\mathbf{A} \in \mathbb{R}^{I\times R}$, $\mathbf{B} \in \mathbb{R}^{J\times R}$, $\mathbf{C} \in \mathbb{R}^{K\times R}$
while (convergence criterion) do
    $\mathbf{Z} = \mathbf{C} \odot \mathbf{B}$
    $\mathbf{A} = \mathbf{X}_{(1)}\mathbf{Z}(\mathbf{Z}'\mathbf{Z})^{-1}$
    $\mathbf{Z} = \mathbf{C} \odot \mathbf{A}$
    $\mathbf{B} = \mathbf{X}_{(2)}\mathbf{Z}(\mathbf{Z}'\mathbf{Z})^{-1}$
    $\mathbf{Z} = \mathbf{B} \odot \mathbf{A}$
    $\mathbf{C} = \mathbf{X}_{(3)}\mathbf{Z}(\mathbf{Z}'\mathbf{Z})^{-1}$
end while
```

2.3.3.2 Tucker3-ALS

Based on Tucker3 model formulation in Equation (2.10), the objective function of the model is:

$$L_2 = ||\mathbf{X}_{(1)} - \mathbf{A}\mathbf{G}_{(1)} \left(\mathbf{C} \otimes \mathbf{B}\right)'||_F^2 \tag{2.12}$$

such that factor matrices **A**, **B**, **C** are column-wise orthogonal. In order to minimize L_2 in ALS form, we minimize for one of the factor matrices at a time and fix others. Algorithm 2 describes `Tucker3-ALS` procedure for fitting a (P, Q, R)-component Tucker3 model to $\underline{\mathbf{X}} \in \mathbb{R}^{I\times J\times K}$. Initialization methods and convergence criterion for PARAFAC-ALS described earlier can also be used for Tucker3-ALS. In Tucker3-ALS algorithm, $SVD(\mathbf{Z}, P)$ denotes the first P left singular vectors of $\mathbf{Z}$. N-mode generalization of this algorithm is also referred as Higher-Order Orthogonal Iteration

(HOOI) [80]. Other algorithms for fitting a Tucker3 model include slice projection by Wang et al. [81] and multislice projection by Turney et al. [82].

Algorithm 2 Tucker3-ALS($\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$,[$P$,$Q$,$R$])

```
Initialize $\mathbf{A} \in \mathbb{R}^{I \times P}$, $\mathbf{B} \in \mathbb{R}^{J \times Q}$, $\mathbf{C} \in \mathbb{R}^{K \times R}$
while (convergence criterion) do
    $\mathbf{Z} = \mathbf{X}_{(1)} (\mathbf{C} \otimes \mathbf{B})$
    $\mathbf{A} = SVD(\mathbf{Z}, P)$
    $\mathbf{Z} = \mathbf{X}_{(2)} (\mathbf{C} \otimes \mathbf{A})$
    $\mathbf{B} = SVD(\mathbf{Z}, Q)$
    $\mathbf{Z} = \mathbf{X}_{(3)} (\mathbf{B} \otimes \mathbf{A})$
    $\mathbf{C} = SVD(\mathbf{Z}, R)$
end while
$\mathbf{G}_{(1)} = \mathbf{A}'\mathbf{X}_{(1)} (\mathbf{C} \otimes \mathbf{B})$
```

2.3.4 Applications of multiway modeling to genomic data

Multiway models and their extensions have been recently used in bioinformatics and genomics [83]. Acar et al. identified epileptic seizures using PARAFAC model on Epilepsy tensor of the form *time samples* × *scales* × *electrodes* [84]. Higher-order singular value decomposition (HOSVD) is used by Omberg et al. on the tensor of the form *gene* × *time* × *condition* for mRNA expression data and found conserved genes and a genome-scale correlation between DNA replication initiation and RNA transcription [85]. Muralidhara et al. applied the same method on *organisms* × *nucleotides* × *positions* tensor from rRNA sequence dataset, and found simultaneous convergent and divergent evolution in rRNA [86]. Yener et al. used PARAFAC and Tucker3 models on the tensor of the form *gene locus link* × *gene ontology category* × *osteogenic stimulant* from human mesenchymal stem cell dataset and revealed that stem cells expressed two distinct, stimulus-dependent sets of functionally related genes [87]. Multiway modeling is also used in other areas such as chemometrics, psychometrics, computer vision, signal processing and social network analysis.

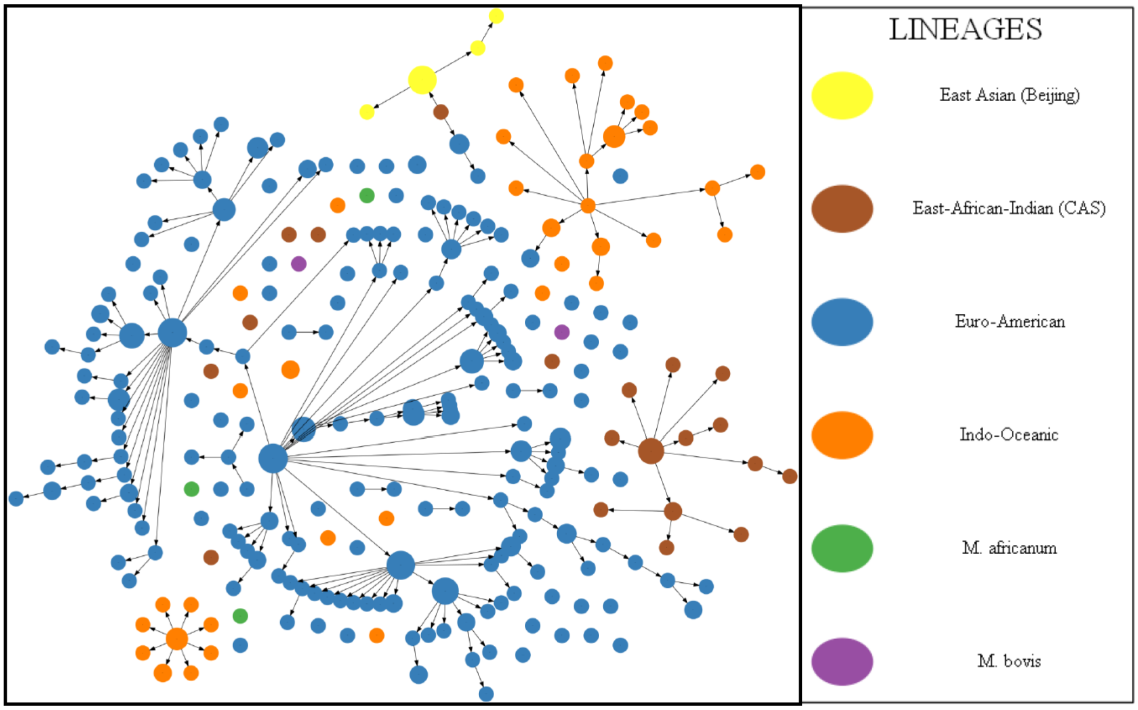


Figure 2.3: The spoligoforest of the patient dataset from NYS-DOH. Each node in the spoligoforest represents a spoligotype, and each edge represents a putative mutation event from ancestor spoligotype to its descendants. Each node is colored by the major lineage of MTBC strains with the associated spoligotype. There are 254 nodes and 185 edges in the spoligoforest.

2.4 Phylogenetic analysis

Phylogenetics is the study of evolution among a set of organisms, also called *taxa*. The inferred evolutionary history of taxa can be represented as a graphical structure called a *phylogenetic tree* [88]. In phylogenetic trees, each leaf node represent a taxon and each internal branch point represents a speciation event. To construct evolutionary relationships between spoligotypes of MTBC, a different graphical structure, called *spoligoforest*, is used [89]. Figure 2.3 shows the spoligoforest of a dataset from New York State Department of Health (NYS-DOH) with 254 nodes and 185 edges. Each node in the spoligoforest, internal or leaf node, represents a spoligotype, and each branch represents a mutation event from the ancestor spoligotype to its descendants. Spoligoforests are phylogenetic trees for spoligotypes, while being structurally different.

Various methods are used to infer phylogenies and to build the corresponding phylogenetic trees. These methods are distance methods, parsimony methods, likelihood methods and Bayesian methods. Next, we briefly review these phylogenetic tree generation methods.

2.4.1 Distance methods

Pairwise distances can be used to build phylogenetic trees. Examples of distance methods include least-squares method, unweighted pair-group method using arithmetic averages (UPGMA) [90], neighbour-joining method [91], and Fitch-Margoliash method [92]. Distance methods are fast and consistent, they produce a single phylogenetic tree. On the other hand, if evolutionary rates vary from taxon to taxon, distance methods are unable to propagate this change [93].

2.4.2 Parsimony methods

Parsimony methods are used to infer phylogenies based on the principle of minimum net amount of evolution [94]. The phylogenetic tree based on maximum parsimony methods are called the *most parsimonious tree*. Examples of parsimony methods include Camin-Sokal parsimony, Dollo parsimony, polymorphism parsimony and Wagner parsimony [93]. It is also possible to weight mutations in parsimony and use weighted parsimony methods. Parsimony methods can also be extended to distance-based methods such as neighbour-joining (NJ) algorithm if there are multiple states of taxa, in which case the distances matter [91].

2.4.3 Likelihood methods

Given a model of evolution, maximum likelihood methods estimate the likelihood of all possible tree topologies which can generate the observed phylogeny [95]. The tree topology with maximum likelihood is assigned as the accurate phylogenetic tree. Likelihood methods are consistent and they can make corrections to phylogeny when evolutionary rates vary. On the other hand, they are computationally expensive.

2.4.4 Bayesian methods

Bayesian methods are similar to likelihood methods, with additional prior probability of tree topology [96]. Most commonly used Bayesian method for phylogeny is MCMC method Metropolis Hastings algorithm. The phylogenies produced by Bayesian methods are highly accurate, but calculating the posterior probability of a tree is time consuming.

CHAPTER 3
SUBLINEAGE STRUCTURE ANALYSIS OF *MYCOBACTERIUM TUBERCULOSIS* COMPLEX STRAINS USING MULTIPLE-BIOMARKER TENSORS

3.1 Introduction and Background

Tuberculosis (TB), a bacterial disease caused by *Mycobacterium tuberculosis* complex (MTBC), is a leading cause of death worldwide. In the United States, isolates from all TB patients are routinely genotyped by multiple biomarkers. The biomarkers include Spacer Oligonucleotide Types (spoligotypes), Mycobacterial Interspersed Repetitive Units - Variable Number Tandem Repeats (MIRU-VNTR), IS*6110* Restriction Fragment Length Polymorphisms (RFLP), Long Sequence Polymorphisms (LSPs), and Single Nucleotide Polymorphisms (SNPs).

Genotyping of MTBC is used to identify and distinguish MTBC into distinct lineages and/or sublineages that are quite useful for TB tracking, TB control, and examining host-pathogen relationships [4]. The six main major lineages of MTBC are *M. africanum*, *M. bovis*, *M. tuberculosis* subgroup Indo-Oceanic, *M. tuberculosis* subgroup Euro-American, *M. tuberculosis* subgroup East Asian (Beijing) and *M. tuberculosis* subgroup East-African Indian (CAS). Other major lineages exist such as *M. canettii* and *M. microti*, but they do not commonly occur in the US, so we do not consider them here. These major lineages can be definitively characterized using LSPs [97], but typically only spoligotypes and MIRU are collected for the purpose

* Portions of this chapter previously appeared as:

C. Ozcaglar, A. Shabbeer, S. Vandenberg, B. Yener, and K. P. Bennett, "Sublineage structure analysis of *Mycobacterium tuberculosis* complex strains using multiple-biomarker tensors," *BMC Genomics*, vol. 12, no. Suppl 2, p. S1, 2011.

C. Ozcaglar, A. Shabbeer, S. Vandenberg, B. Yener, and K. P. Bennett, "Examining the sublineage structure of *Mycobacterium tuberculosis* complex strains with multiple-biomarker tensors," in *Proc. 2010 IEEE Int. Conf. Bioinformatics and Biomedicine (BIBM)*, Hong Kong, pp. 543-548, 2010.

C. Ozcaglar, A. Shabbeer, S. L. Vandenberg, B. Yener, and K. P. Bennett, "A clustering framework for *Mycobacterium tuberculosis* complex strains using multiple-biomarker tensors," Dept. Comp. Sci., Rensselaer Polytechnic Inst., Troy, NY, Tech. Rep. 10-08, 2010.

of TB surveillance. Classification, similarity search, and expert-rule based methods have been developed to correctly map isolates genotyped using MIRU and/or spoligotypes to the major lineages [18, 20, 43].

While sublineages of MTBC are routinely used in the TB literature, their exact definitions, names, and numbers have not been clearly established. The SpolDB4 database contains 39,295 strains and their spoligotypes with the vast majority of them labeled and classified into 62 sublineages [27], but many of these are considered to be "potentially phylogeographically-specific MTBC genotype families", rather than distinct phylogenetic sublineages with known biomarkers. Therefore, further analysis is needed to confirm these sublineages. The highly-curated MIRU-VNTR*plus* website, which focuses primarily on MIRU, defines 22 sublineages. New definitions of sublineages based on LSPs and SNPs are being discovered; e.g. the RD724 polymorphism corresponds to the previously defined SpolDB4 T2 sublineage, also known as the Uganda strain in MIRU-VNTR*plus* [98]. Now large databases using spoligotype, MIRU patterns, and RFLP exist. The United States Centers for Disease Control and Prevention (CDC) has gathered spoligotypes and MIRU isolates for over 37,000 patients. Well-defined TB sublineages based on spoligotype and MIRU are critical for both TB control and TB research.

The goal of this paper is to examine the sublineage structure of MTBC on the basis of multiple biomarkers. The proposed method reveals structure not captured in SpolDB4 spoligotype families because SpolDB4 sublineage only take into account a single biomarker, spoligotypes. A spoligotype-only tool, SPOTCLUST, was used to find MTBC sublineages using an unsupervised probabilistic model, reflecting spoligotype evolution [44]. A key issue is to combine spoligotype and MIRU into a single unsupervised learning model. When MIRU patterns are considered, SpolDB4 families that are well-supported by spoligotype signatures may become ambiguous, or allow subdivision/merging of the families. Existing phylogenetic methods can be readily applied to MIRU patterns, but specialized methods are needed to accurately capture how spoligotypes evolve. It is not known how to best combine spoligotype and MIRU patterns to infer a phylogeny. The online tool www.MIRUVNTRplus.org determines lineages by using similarity search to a labeled database. The user must select the distance measure which is defined using spoligotypes and/or MIRU patterns, possibly yielding different results.

In this study, we develop a tensor clustering framework to find the sublineage structure of MTBC strains labeled by major lineages based on multiple biomarkers. This is an unsupervised learning problem. We generate multiple-biomarker tensors of MTBC strains for each major lineage and apply multiway models for dimensionality reduction. The model accurately captures spoligotype evolutionary dynamics using contiguous deletions of spacers. The tensor transforms spoligotypes and MIRU into a new representation, where traditional clustering methods apply without users having to decide *a priori* how to combine spoligotype and MIRU patterns. Strains are clustered based on the transformed data without using any information from SpolDB4 families. Clustering results lead to the subdivision of major lineages of MTBC into groups with clear and distinguishable spoligotype and MIRU signatures. Comparison of the tensor sublineages with SpolDB4 families suggests dividing or merging some SpolDB4 families. As a way of validating multiple-biomarker tensors, we use them in a supervised learning model to predict major lineages using spoligotype deletions and MIRU. We compare the prediction accuracy of the multiple-biomarker tensor model created with N-PLS (N-way partial least squares) with the 2-way PLS applied to matrix data and an existing conformal Bayesian Network approach.

In the next section, we give a brief background on clustering and multiway analysis of post-genomic data, spoligotyping, and MIRU typing.

3.1.1 Clustering post-genomic data

Data clustering is a class of techniques for unsupervised classification of data samples into groups of similar behavior, function, or trait [99]. Clustering can be used in post-genomic data analysis to group strains with similar traits. It is common practice to use different clustering methods and use *a priori* biological knowledge to interpret the clusters, but computational cluster validation is needed to validate results without prior knowledge for unsupervised classification. A great survey by Handl et al. outlines the steps of computational cluster analysis on post-genomic data [100]. An application of computational cluster validation on microarray data by Giancarlo et al. compares the results of clusterings using various cluster validation indices [101]. Eisen et al. clusters gene expression data which groups genes of similar functions [42]. Improved clustering techniques have been developed, but how to combine multiple sources of information in one clustering is an open question.

3.1.2 Application of multiway models to post-genomic data clustering

Clustering on post-genomic data can be accomplished based on multiple sources of ground truth. The ground truth can be based on multiple biomarkers, host and pathogen, or antigen and antibody. A survey by Kriegel et al. outlines the methods for finding clusters in high-dimensional data [102]. Analysis of multiway arrays for data mining is frequently used today in various fields, including bioinformatics, to use multiple sources of prior information simultaneously [83]. Alter and Golub use higher-order eigenvalue decomposition on a *networks* × *genes* × *genes* tensor and find significant subnetworks associated with independent pathways in a genome-scale network of relations among all genes of cellular systems [103]. Omberg et al. use higher-order singular value decomposition on DNA microarray data, obtaining the core tensor of *eigenarrays* × *x-eigengenes* × *y-eigengenes* and finding correlation between genomes in the subtensors of the core tensor [85]. Multiway analysis of EEG data identifies epileptic seizures [84]. Use of common partitive and hierarchical clustering algorithms accompanied with multiway modeling of high-dimensional data finds functionally related genes in stem cells [87]. Similarly, multiple biomarkers of the MTBC genome can be used to cluster MTBC strains.

3.1.3 Spoligotyping

Spoligotyping is a DNA fingerprinting method that exploits the polymorphisms in the direct repeat (DR) region of the MTBC genome. The DR region is a polymorphic locus in the genome of MTBC which consists of direct repeats (36 bp), separated by unique spacer sequences of 36 to 41 bp [17]. The method uses 43 spacers, thus a spoligotype is typically represented by a 43-bit binary sequence. Zeros and ones in the sequence correspond to the absence and presence of spacers respectively. Mutations in the DR region involve deletion of one or more contiguous spacers. To capture this mechanism of mutation in our model, we find informative contiguous spacer deletions and represent spoligotype deletions as a binary vector, where one indicates that a specific contiguous deletion occurs (i.e. a specified contiguous set of spacers are all absent) and zero means at least one spacer is present in that contiguous set of spacers.

Large datasets of MTBC strains genotyped by spoligotype have been amassed such as SpolDB4 [27] and a more extended online version SITVIT (http://www.pasteur-guadeloupe.fr:8081/SITVITDemo/index.jsp). Spoligotypes can be readily used to identify commonly accepted major lineages of MTBC with high accuracy [20]. SpolDB4 defined a set of phylogeographic sublineages or families based on expert derived rules that are in common use in the TB community. In contrast to the major lineages that have been validated by more definitive markers such as single nucleotide polymorphisms and long sequence polymorphism, the exact definition of MTBC sublineages and the accuracy of the SpolDB4 families created only using spoligotypes remain open questions.

3.1.4 MIRU-VNTR typing

MIRU is a homologous 46-100 bp DNA sequence dispersed within intergenic regions of MTBC, often as tandem repeats. MIRU-VNTR typing is based on the number of tandem repeats of MIRUs at certain identified loci. Among these 41 identified mini-satellite regions on the MTBC genome, different subsets of sizes 12, 15, and 24 are proposed for the standardization of MIRU-VNTR typing [18]. In this study, we use 12 MIRU loci for genotyping MTBC. Thus, the MIRU pattern is represented as a vector of length 12, each entry representing the number of repeats in each MIRU locus.

3.2 Methods

3.2.1 Datasets

The dataset comprises 6848 distinct MTBC strains as determined by spoligotype and 12-loci MIRU, labeled with major lineages and SpolDB4 families. The strains are mainly from the CDC dataset - a database collected by the CDC from

2004-2008 labeled with the major lineages collected by the TB-Insight project (http://tbinsight.cs.rpi.edu/) that was previously studied in [20]. We also used the MIRU-VNTR*plus* dataset from www.MIRUVNTRplus.org which is labeled with SpolDB4 lineages and sublineages. The original SpolDB4 labeled dataset provided in an online supplement [27] contains only spoligotypes. We found all occurrences of these spoligotypes in the CDC and MIRU-VNTR*plus* dataset and constructed a database with spoligotype and MIRU patterns, with major lineages as determined by CDC, and sublineages as given in the SpolDB4 database [27]. The numbers of strains for each major lineage in the resulting dataset are shown in Table 3.1. We created 6 datasets from the CDC+MIRU-VNTR*plus* dataset, one for each major lineage. These same 6 major lineage datasets were merged into one for the supervised learning experiment.

3.2.2 TCF: Tensor Clustering Framework

Clustering MTBC strains based on multiple-biomarker tensors consists of a sequence of steps. First, we find informative feature set of spoligotype deletions and generate a tensor. Second, we apply multiway models on the tensor and get a score matrix for the strain mode. Third, we use this score matrix to determine the similarity between strains, and cluster them using a stable version of k-means. In the final step, we evaluate the clustering results using cluster validity indices. This stepwise Tensor Clustering Framework (TCF) is outlined in Figure 3.1. The software for TCF is available at http://sourceforge.net/projects/tcff/. We describe the steps of the tensor clustering framework in this section.

3.2.2.1 Feature Selection and Tensor Generation

Feature Selection The spoligotype pattern captures the variability in the DR locus of the MTBC genome. A spoligotype consists of 43 spacers represented as a 43-bit binary sequence, and according to the hidden parent assumption, one or more contiguous spacers can be lost in a deletion event, but rarely gained [44, 104]. Therefore, there are $\sum_{i=1}^{43} i = 946$ possible deletions of lengths varying from 1 to 43 in a spoligotype. Only subsets of spoligotype deletions are required for effective

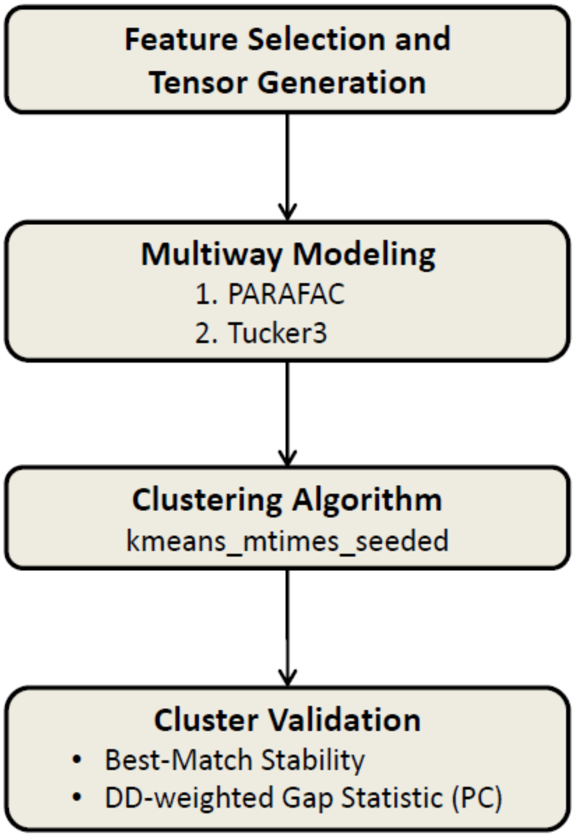


Figure 3.1: Tensor clustering framework for MTBC strains. High-dimensional genotype data is decomposed into two-dimensional arrays using multiway models, which are then used as input to the `kmeans_mtimes_seeded` algorithm. Clusterings are validated using best-match stability. In case of a tie, the DD-weighted gap statistic is used to pick the number of clusters.

Table 3.1: Numbers of strains in each major lineage of CDC+MIRU-VNTR*plus* dataset and numbers of spoligotype deletions identified by the feature selection algorithm.

Major lineage	# Strains	# Spoligotype deletions
M. africanum	64	22
M. bovis	102	34
East Asian (Beijing)	571	5
East-African Indian(CAS)	508	18
Indo-Oceanic	1023	28
Euro-American	4580	109

discrimination of MTBC strains. A set of 12 deletion sequences of spoligotypes reported by Shabbeer et al. have proven to be good discriminator spacer deletions for major lineage classification [26]. These 12 deletion sequences are used in the supervised learning study. Another set of 81 deletion sequences of spoligotypes reported by Brudey et al. have proven to be good discriminator spacer deletions for SpolDB4 sublineage classification [27].

Within the TCF, we built a feature selection algorithm to find spacer deletions that are informative. This insures that the results are not biased by *a* priori selection of spoligotype deletions. Given a set of spoligotypes, we first calculate the frequency f_i, $i = 1, .., 946$, of each possible deletion among the spoligotypes of strains. If $f_i = 1$, the deletion is a common deletion. If $0 \leq f_i < threshold$, the deletion is a nonexistent deletion, where $threshold$ is data dependent and $threshold = 0.05$ is used by default. The deletions with frequency f_i such that $threshold \leq f_i < 1$ are uncommon deletions. In the second step, we iterate through the set of uncommon deletions U, and remove an uncommon deletion $u \in U$, if there exists a common deletion $c \in C$ which is a substring of u. We assign the final set of uncommon deletions as the feature set. Using the final feature set, we determine spoligotype deletions that are effective in discriminating the strains of the dataset. Algorithm 3 summarizes the feature selection procedure. Numbers of spoligotype deletions for each major lineage, found informative by the feature selection algorithm, are given in Table 3.1.

Algorithm 3 `FeatureSelection(Spoligotypes, th)`

```
// Classify all possible spoligotype deletions according to their frequency $f_i$
        - $0 \leq f_i < th$: Nonexistent deletions (N)
        - $th \leq f_i < 1$: Uncommon Deletions (U)
        - $f_i = 1$: Common deletions (C)
where $th$ is the upper bound of frequency for nonexistent deletions.
// Remove uncommon deletions which are a superset of a common deletion
for each uncommon deletion $u \in U$ do
    if $\exists c \in C$ which is a substring of $u$ then
        Remove $u$ from uncommon deletions: $U = U \setminus \{u\}$
    end if
end for
Return uncommon deletion set U as the final feature set.
```

Tensor Generation We generated multiple-biomarker tensors using two biomarkers, spoligotype deletions and MIRU patterns. The spoligotype deletions found informative by the feature selection algorithm are used in the generation of multiple-biomarker tensors. The strain dataset is arranged as a three-way array with strains in the first mode, spoligotype deletions in the second mode, and MIRU patterns in the third mode. Each entry $\underline{\mathbf{X}}(i,j,k)$ in the tensor corresponds to the number of repeats in MIRU locus k of strain i with spoligotype deletion j. If spoligotype deletion j does not exist in strain i, then the tensor entry $\underline{\mathbf{X}}(i,j,.)$ is 0. Thus, strain datasets are formed as *Strains* × *Spoligotype deletions* × *MIRU patterns* tensors, as shown in Figure 3.2. Mathematically, each strain is represented as the outer product of the binary spoligotype deletion vector and the MIRU pattern vector, which results in a biomarker kernel matrix. Biomarker kernel matrices of the same size for each strain form the multiple-biomarker tensor. Generation of the multiple-biomarker tensor from biomarkers of each strain is shown in Figure 3.3. We represent spoligotype deletions with a binary vector $\vec{s}$, where $s_i \in \{0,1\}$, $i \in \{1,..,n\}$, and n is the number of informative spoligotype deletions found using the feature selection algorithm, detailed in the methods section. We represent 12-loci MIRU with a digit vector $\vec{m}$, where $m_j \in \{1,..,9,>9\}$ and $j \in \{1,..,12\}$. The entries of the multiple-biomarker tensor which combines spoligotype and MIRU information can be formulated as:

$$\underline{\mathbf{X}}_{ijk} = \delta_{ij}\ r_{ik}$$

where

$$\delta_{ij} = \begin{cases} 0, & \text{if spoligotype deletion } j \text{ does not occur in strain } i, \\ 1, & \text{if spoligotype deletion } j \text{ occurs in strain } i. \end{cases}$$

and r_{ik} is the number of repeats in MIRU locus k of strain i.

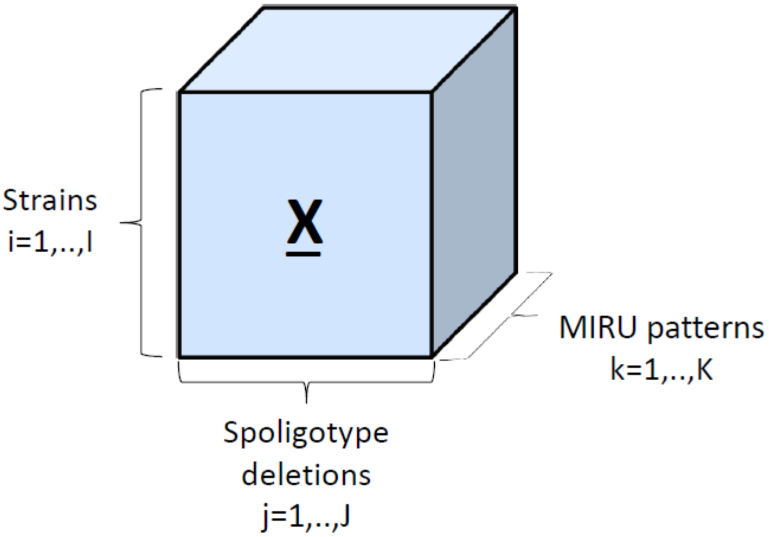


Figure 3.2: *Strains* × *Spoligotype deletions* × *MIRU patterns* tensor. Each entry $\underline{\mathbf{X}}(i,j,k)$ of the tensor represents the number of repeats in MIRU locus k of strain i with spoligotype deletion j.

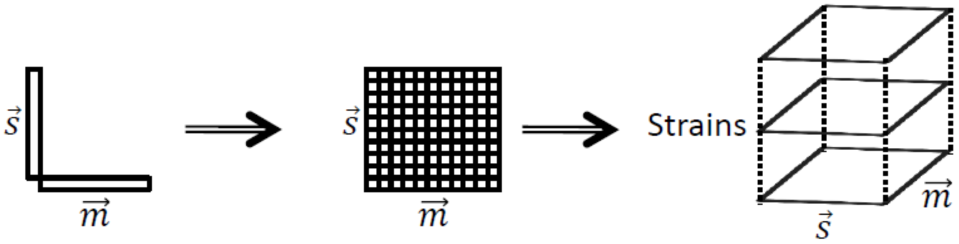


Figure 3.3: Biomarker kernel matrix $\vec{s}\otimes\vec{m}$ for each strain forms multiple-biomarker tensor. Vector $\vec{s}$ represents spoligotype deletions and $\vec{m}$ represents MIRU patterns.

3.2.2.2 Multiway modeling

Multiway models are needed to fit a model to multiway arrays. We used PARAFAC and Tucker3 techniques to model the tensors. We determined the number of components for each model to ensure a bound on the explained variance of data.

Multiway models We used PARAFAC and Tucker3 models to explain the tensor with high accuracy. Multiway modeling of tensors was carried out using the *n-way Toolbox* of MATLAB by Bro et al. and the *PLS toolbox* [105, 106].

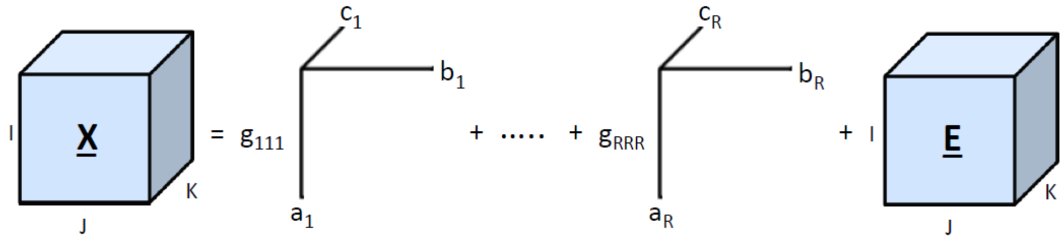


Figure 3.4: PARAFAC model of a three-way array with R components. The tensor is modeled as a linear combination of rank-one tensors for each mode.

PARAFAC PARAFAC is a generalization of singular value decomposition to multiway data [69, 107]. A 3-way array $\underline{\mathbf{X}} \in \mathbb{R}^{I\times J\times K}$ is modeled by an R-component PARAFAC model as follows:

$$\underline{\mathbf{X}}_{ijk} = \sum_{r=1}^{R} \underline{\mathbf{G}}_{rrr}\mathbf{A}_{ir}\mathbf{B}_{jr}\mathbf{C}_{kr} + \underline{\mathbf{E}}_{ijk}$$

where $\mathbf{A} \in \mathbb{R}^{I\times R}$, $\mathbf{B} \in \mathbb{R}^{J\times R}$, $\mathbf{C} \in \mathbb{R}^{K\times R}$ are component matrices of first, second, and third mode. $\underline{\mathbf{G}} \in \mathbb{R}^{R\times R\times R}$ is the core array, and $\underline{\mathbf{E}} \in \mathbb{R}^{I\times J\times K}$ is the residual term containing all unexplained variation. A description of the PARAFAC model is shown in Figure 3.4.

The PARAFAC model is symmetric in all modes and the number of components in each mode is the same [73]. The PARAFAC model is a simple model, which comes with a restriction of the equality on the number of components in each mode which makes it difficult to fit a data array with the PARAFAC model. One advantage of the PARAFAC model is its uniqueness: fitting the PARAFAC model with the same number of components to a given multiway dataset returns the same result.

Tucker3 Tucker3 is an extension of bilinear factor analysis to multiway datasets [72]. A 3-way array $\underline{\mathbf{X}} \in \mathbb{R}^{I\times J\times K}$ is modeled by a (P, Q, R)-component Tucker3 model as follows:

$$\underline{\mathbf{X}}_{ijk} = \sum_{p=1}^{P}\sum_{q=1}^{Q}\sum_{r=1}^{R} \underline{\mathbf{G}}_{pqr}\mathbf{A}_{ip}\mathbf{B}_{jq}\mathbf{C}_{kr} + \underline{\mathbf{E}}_{ijk}$$

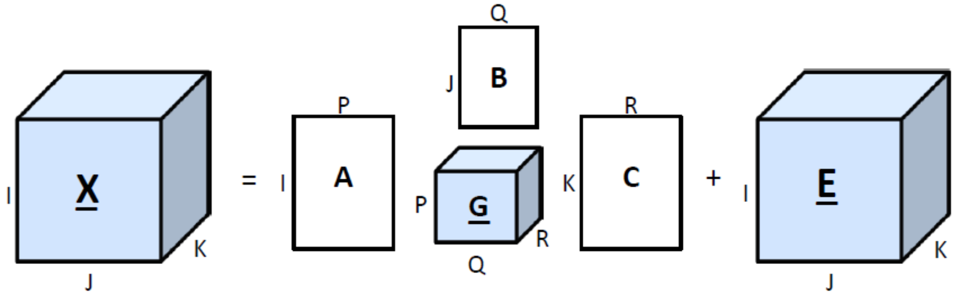


Figure 3.5: Tucker3 model of a three-way array with (P,Q,R) components at each mode.

where $\mathbf{A} \in \mathbb{R}^{I\times P}$, $\mathbf{B} \in \mathbb{R}^{J\times Q}$, $\mathbf{C} \in \mathbb{R}^{K\times R}$ are the component matrices of first, second and third modes respectively. $\underline{\mathbf{G}} \in \mathbb{R}^{P\times Q\times R}$ is the core array and $\underline{\mathbf{E}} \in \mathbb{R}^{I\times J\times K}$ is the residual term. A description of the Tucker3 model is shown in Figure 3.5.

Tucker3 is a more flexible model compared to PARAFAC. This flexibility is due to the core array $\underline{\mathbf{G}}$, which allows interaction of any factor in a mode with any other factor in other modes [65]. Therefore, the number of components for each mode can be different. This results in indeterminacy of the Tucker3 model, since it cannot determine the component matrices uniquely.

Model validation A multiway model is appropriate if adding more components to any mode does not improve the fit considerably. There is a tradeoff between the complexity of the model and the variance of the data explained by the model. Therefore, validation of a model also determines a suitable complexity for the model. We used the core consistency diagnostic (CORCONDIA) to determine the number of components of the PARAFAC model [108]. The core consistency diagnostic measures the similarity of the core array $\underline{\mathbf{G}}$ of the model and the superdiagonal array of ones. Core consistency is always less than or equal to 100% and may also be negative. As a rule of thumb, Bro et al. suggests that a core consistency above 90% implies a trilinear model [108]. In our experiments, we kept core consistency above 90%, while still explaining the variance of the data as much as possible with a trilinear model. We determined the number of components of the Tucker3 model by rank reduction on the unfolded tensor along each mode, and these components explain over 90% of the variance of the data.

3.2.2.3 Clustering algorithm

We developed the `kmeans_mtimes_seeded` algorithm, a modified version of the k-means algorithm, to group MTBC strains based on the score matrices of the multi-way models. `K-means` is a commonly used clustering algorithm with two weaknesses: 1) Initial centroids are chosen randomly, 2) The objective value of `k-means`, measured as within-cluster sum of squares, may converge to local minima, rather than finding the global minimum. We solve these problems with two improvements: 1) Initial centroids are chosen by careful seeding, using a heuristic called `kmeans++`, suggested by Arthur et al. [109]. Let $D(x)$ represent the shortest Euclidean distance from data point x to the closest center already chosen. `kmeans++` chooses a new centroid at each step such that the new centroid is furthest from all chosen centroids. Algorithm 4 summarizes the `kmeans++` procedure. 2) The local minima problem is partially solved by repeating the `k-means` algorithm multiple times and retrieving the run with the minimum objective value. We repeated the algorithm $m = 20$ times. The `kmeans_mtimes_seeded` algorithm combines these two improvements, as summarized in algorithm 5. The `kmeans_mtimes_seeded` algorithm is more stable compared to the `k-means` algorithm, and produces more accurate clusters.

Algorithm 4 `kmeans++(A,k)`

```
Pick the first centroid $c_1$ at random: InitCentroids = $\{c_1\}$
for $i = 2$ to $k$ do
    Find $D(a)$, distance to the closest centroid picked so far, for each data point $a \in$ A
    Pick the data point $a$ with maximum $D(a)$ as new centroid
            $c_i = \arg\max_a D(a)$
    Add $c_i$ to the set of initial centroids:
            InitCentroids = InitCentroids $\cup \{c_i\}$
end for
Run kmeans(A,k) with InitCentroids
```

3.2.2.4 Cluster validation

Clustering results for the MTBC strains are evaluated to determine the best choice for the number of clusters and compare the chosen clustering with existing sublineages using cluster validity indices. We used the best-match stability to pick

Algorithm 5 `kmeans_mtimes_seeded(A,k,m)`

```
for $i = 1$ to $m$ do
    kmeans++(A,k)
    Get the objective value of k-means run $i$
end for
Pick the k-means run with the minimum objective value
```

the most stable clusterings. In case of a tie in average best-match stability, we used the DD-weighted gap statistic for cluster validation [110]. We compare our clusters to an existing classification using the F-measure.

Best-Match Stability The stability of a clustering is measured by the distribution of pairwise similarities between clusterings of subsamples of the data. The idea behind stability is that if we repeatedly sample data points and apply the same clustering algorithm to the subsample, then an effective clustering algorithm applied to well separated data should produce clusterings that do not vary much for different subsamples [111]. In such cases, the algorithm is stable independent of input randomization. We use best-match stability as suggested by Hopcroft et al. [112] to assess stability. The algorithm clusters the same data multiple times, and compares the reference cluster to model clusterings. We used 25 model clusterings to compare with the reference cluster. The stability of each cluster is calculated by finding the average best match between this cluster and the clusters identified using other model clusterings. High average best-match values denote that the two clusters have many strains in common and are of roughly the same size [44]. We also calculate the average best-match of a clustering by finding the average of best-match values for all clusters in the reference clustering. Best-match stability of a cluster C, compared to a model clustering $Cref = \bigcup_{i=1}^{k} refC_i$, is calculated as:

$$best\_match\left(C, \bigcup_{i=1}^{k} refC_i\right) = \max_{i=1,..,k} match(C, refC_i)$$

where

$$match(C, C^{'}) = \frac{\mid C \cap C^{'} \mid}{\max\left(\mid C \mid, \mid C^{'} \mid\right)}$$

and $refC_i$ is the set of items in reference cluster i.

DD-Weighted Gap Statistic (PC) Tibshirani et al. proposed a cluster validity index called the gap statistic, which is based on the within-cluster sum of squares (WCSS) of a clustering [113]. Let the dataset be $\mathbf{X} \in \mathbb{R}^{n \times p}$ consisting of n data points with p dimensions. Let d_{ij} be the Euclidean distance between data points i and j. After clustering this dataset, suppose that we have k clusters C_1, .., C_k, where C_i denotes the indices of data points in cluster i, of size $n_i =\mid C_i \mid$. The sum of within-cluster pairwise distances for cluster r is defined as:

$$D_r = \sum_{i,j \in C_r} d_{ij}$$

and the within-cluster sum of squares for a clustering is defined as:

$$W_k = \sum_{r=1}^{k} D_r \, .$$

The idea of the gap statistic method is to compare W_k and its expected value under a reference distribution of the dataset. Therefore, the gap value is defined as:

$$Gap_n(k) = E_n^*\{log(W_k)\} - log(W_k)$$

where E_n^* represents the expected value under a sample of size n based on a reference distribution. The optimal number of clusters is the value $\hat{k}$ for which $Gap_n(k)$ is maximized. The selection of number of clusters via gap statistic is summarized in [113].

The reference distribution can be one of two choices: uniform distribution (Gap/Unif), or a uniform distribution over a box aligned with the principal components of the dataset (Gap/PC). Experiments by Tibshirani et al. show that Gap/PC finds the number of clusters more accurately, therefore we used Gap/PC in this study [113].

The gap statistic is a powerful method for estimating the number of clusters in a dataset. However, a study by Dudoit et al. showed that the gap statistic does not estimate the correct number of clusters for every case [114]. This may be because

W_k increases as the number of data points increases. Hierarchical structure of the data may also cause problems. The data may be composed of nested clusters and the gap statistic will be capturing only the minimum of these candidate numbers of clusters. Yan et al. suggested a 2-step improvement to the gap statistic, called the DD-weighted gap statistic [115]. They defined average within-cluster pairwise distances for cluster r as follows:

$$\overline{D}_r = \frac{D_r}{2n_r(n_r-1)}$$

and the weighted within-cluster sum of squares $\overline{W_k}$ as:

$$\overline{W_k} = \sum_{r=1}^{k} \overline{D}_r = \sum_{r=1}^{k} \frac{D_r}{2n_r(n_r-1)}\,.$$

Based on $\overline{W_k}$, the weighted gap statistic $\overline{Gap}_n(k)$ is defined as:

$$\overline{Gap}_n(k) = E_n^*\{log(\overline{W_k})\} - log(\overline{W_k})\,.$$

Let $D\overline{Gap}_n(k)$ denote the difference in $\overline{Gap}_n(k)$ when the number of clusters is raised from k-1 to k. $D\overline{Gap}_n(k)$ is defined as:

$$D\overline{Gap}_n(k) = \overline{Gap}_n(k) - \overline{Gap}_n(k-1)\,.$$

$D\overline{Gap}_n(k) > 0$ for $k < \hat{k}$, and otherwise it will be close to zero. Therefore, to find a “knee” point in the plot, they introduce a second difference equation and define $DD\overline{Gap}_n(k)$ as:

$$\begin{aligned} DD\overline{Gap}_n(k) &= D\overline{Gap}_n(k) - D\overline{Gap}_n(k+1) \\ &= 2\overline{Gap}_n(k) - \overline{Gap}_n(k-1) - \overline{Gap}_n(k+1)\,. \end{aligned}$$

$DD\overline{Gap}_n(k)$ is maximized when k is equal to the true number of clusters. The advantage of $DD\overline{Gap}_n(k)$ over the gap statistic is that there may be multiple peaks in the plot of $DD\overline{Gap}_n(k)$ and this may indicate a hierarchical structure in the data.

Table 3.2: Contingency table of a clustering, where rows represent true classes and columns represent found clusters. Given n data points in the dataset, $a + b + c + d = \binom{n}{2}$.

	Same cluster	Different clusters
Same class	a	b
Different classes	c	d

In such cases, multilayer analysis should be used instead of a single step procedure.

F-measure The F-measure is a weighted combination of precision and recall of a clustering. Since the F-measure combines precision and recall of clustering results, it has proven to be a successful metric. We use the F-measure to evaluate how similar the tensor sublineages are to the SpolDB4 families. According to the contingency table in Table 3.2, precision, recall, and F-measure are defined as:

$$P = \frac{a}{a+c}$$
$$R = \frac{a}{a+b}$$
$$F = \frac{2PR}{P+R}.$$

3.2.3 Multiway Partial Least Squares Regression (N-PLS)

N-PLS is a multiway regression method where at least one of the independent and dependent blocks has at least three modes created by Bro et al. by generalizing PLS to multiway data [116]. Consider independent variables in the X-block, $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$, and dependent variables in the Y-block, $\mathbf{Y} \in \mathbb{R}^{I \times M}$. In our experiments, the X-block is a three-way array and the Y-block is a two-way array. The multiway array $\underline{\mathbf{X}}$ is decomposed using a matricized version $\mathbf{X} \in \mathbb{R}^{I \times JK}$ as:

$$\mathbf{X} = \mathbf{t}\left(\mathbf{w^K} \otimes \mathbf{w^J}\right)' + \mathbf{E} \tag{3.1}$$

and the two-way array $\mathbf{Y}$ is decomposed as:

$$\mathbf{Y} = \mathbf{u}\mathbf{q}' + \mathbf{F} \tag{3.2}$$

where $\mathbf{t} \in \mathbb{R}^{I\times 1}$ and $\mathbf{u} \in \mathbb{R}^{I\times 1}$ are score vectors of $\mathbf{X}$ and $\mathbf{Y}$. $\mathbf{w^J} \in \mathbb{R}^{J\times 1}$ and $\mathbf{w^K} \in \mathbb{R}^{K\times 1}$ are the loading vectors (weights) of the second and third modes of $\underline{\mathbf{X}}$ respectively. $\mathbf{q} \in \mathbb{R}^{M\times 1}$ is the loading vector of $\mathbf{Y}$. $\mathbf{E} \in \mathbb{R}^{I\times JK}$ and $\mathbf{F} \in \mathbb{R}^{I\times M}$ are the residuals of $\mathbf{X}$ and $\mathbf{Y}$ respectively.

Notice that the two-way array $\mathbf{Y}$ is decomposed into one score and one loading vector, whereas the matricized three-way array $\mathbf{X}$ is decomposed into one score and two loading vectors, $\mathbf{w^J}$ and $\mathbf{w^K}$. This is the main difference between N-PLS and PLS. At each iteration of N-PLS, a new PLS component is added. If n PLS components are used, $\mathbf{X}$ is decomposed into component matrices $\mathbf{T} \in \mathbb{R}^{I\times n}$, $\mathbf{W^J} \in \mathbb{R}^{J\times n}$, $\mathbf{W^K} \in \mathbb{R}^{K\times n}$, and $\mathbf{Y}$ is decomposed into component matrices $\mathbf{U} \in \mathbb{R}^{I\times n}$, $\mathbf{Q} \in \mathbb{R}^{M\times n}$.

The aim of N-PLS is to maximize the covariance of $\underline{\mathbf{X}}$ and $\mathbf{Y}$. For this purpose, we define an inner relation linking the $\underline{\mathbf{X}}$ and $\mathbf{Y}$ blocks, using their score matrices, $\mathbf{T}$ and $\mathbf{U}$:

$$\mathbf{U} = \mathbf{T}\mathbf{B} + \mathbf{E_u}\,. \tag{3.3}$$

This requires finding loading vectors $\mathbf{w^J}$ and $\mathbf{w^K}$ such that the covariance of $\mathbf{t}$ and $\mathbf{y}$ are maximized:

$$\begin{aligned} A &= \max_{\mathbf{w^J},\mathbf{w^K}} \left[cov(t,y) \middle| \min \left(\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K} \left(x_{ijk} - t_i {w_j}^J {w_k}^K\right)^2 \right) \right] \\ &= \max_{\mathbf{w^J},\mathbf{w^K}} \left(\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K} y_i x_{ijk} {w_j}^J {w_k}^K \right) \\ &= \max_{\mathbf{w^J},\mathbf{w^K}} \left(\sum_{j=1}^{J}\sum_{k=1}^{K} z_{jk} {w_j}^J {w_k}^K \right) \end{aligned}$$

where $\mathbf{Z} \in \mathbb{R}^{J\times K}$ is a matrix with $z_{jk} = \sum_{i=1}^{I} y_i x_{ijk}$ and $\left|\left|w^J\right|\right| = \left|\left|w^K\right|\right| = 1$. To maximize this expression, we write it in matrix notation:

$$A = \max_{\mathbf{w^J},\mathbf{w^K}} \left(\left(\mathbf{w^J}\right)' \mathbf{Z}\mathbf{w^K} \right) \Rightarrow (\mathbf{w^J}, \mathbf{w^K}) = SVD(\mathbf{Z})\,.$$

The problem of finding $\mathbf{w^J}$ and $\mathbf{w^K}$ is simply solved by SVD on $\mathbf{Z}$ [77,116]. $\mathbf{w^J}$ and $\mathbf{w^K}$ are first left and right singular vectors of $\mathbf{Z}$. To reconstruct $\mathbf{Y}$, we substitute (3.3) in Equation (3.2):

$$\begin{aligned}
\mathbf{Y} &= (\mathbf{TB} + \mathbf{E_u})\,\mathbf{Q}' + \mathbf{F} \\
\mathbf{Y} &= \mathbf{TBQ}' + \mathbf{E_u}\mathbf{Q}' + \mathbf{F} \\
\mathbf{Y} &= \mathbf{TBQ}' + \mathbf{F}^* \qquad (3.4)
\end{aligned}$$

Given $\underline{\mathbf{X}}$ and its decomposition matrices, we can make predictions for a new X-block, using equation 3.4. The derivation of the full and closed predictions with N-PLS has been presented by Smilde et al. [117]. Three alternative methods are proposed by De Jong et al. for derivation of training models via regression coefficients [118]. Bro et al. proposed an improved N-PLS method with better fit of the independent data, keeping regression coefficients and predictions the same [119].

The N-PLS model of a multiway array is a multilinear model, like PARAFAC, which means that it has no rotational freedom. Therefore, the N-PLS model of a multiway array is unique. In this study, we used a 3-way array as the X-block and a 2-way array as the Y-block, therefore we are particularly working on the Tri-PLS2 version of N-PLS, which is summarized in Algorithm 6. The term $\underline{\mathbf{X}}_{(1)}$ in the algorithm refers to $\underline{\mathbf{X}}$ matricized along the first mode. The X-block and Y-block are centered and scaled prior to application of the algorithm. The preprocessing and postprocessing of both X-block and Y-block are done according to centering and scaling methods explained in [120].

Algorithm 6 Tri-PLS2($\underline{\mathbf{X}} \in \mathbb{R}^{I\times J\times K}$, $\mathbf{Y} \in \mathbb{R}^{I\times M}$, N)

```
$\mathbf{X}_0 = \underline{\mathbf{X}}_{(1)}$
$y_0 = \mathbf{Y}(:,1)$
for $i = 1$ to $N$ do
    repeat
        Calculate matrix $\mathbf{Z} \in \mathbb{R}^{J\times K}$ such that $z_{jk} = \sum_{l=1}^{I} y_l x_{ljk}$
        Compute SVD of $\mathbf{Z}$: $\mathbf{Z} = \mathbf{U\Sigma V}'$
        Calculate loading vectors as first left and right singular vectors of $\mathbf{Z}$:
            $\mathbf{w^J} = \mathbf{U}(:,1)$        $\mathbf{w^K} = \mathbf{V}(:,1)$
        Calculate score vector $\mathbf{t}$
            $\mathbf{t} = \mathbf{T}(:,i) = \mathbf{X}_{i-1}\left(\mathbf{w^K}\otimes\mathbf{w^J}\right)$
        $\mathbf{q} = \mathbf{Y}'\mathbf{T}/\mid \mathbf{Y}'\mathbf{T}\mid$
        $\mathbf{y_{i-1}} = \mathbf{Yq}$
    until $\mathbf{y_{i-1}}$ converges
    Calculate regression coefficient $\mathbf{b}$:
        $\mathbf{b} = \mathbf{B}(:,\mathbf{i}) = \left(\mathbf{T}'\mathbf{T}\right)^{-1}\mathbf{T}'\mathbf{y_{i-1}}$
    Deflate $\mathbf{X}$ and $\mathbf{Y}$
        $\mathbf{X}_i = \mathbf{X}_{i-1} - \mathbf{t}\left(\mathbf{w^K}\otimes\mathbf{w^J}\right)'$
        $\mathbf{Y} = \mathbf{Y} - \mathbf{Tbq}'$
end for
```

3.3 Results

Multiple biomarkers of the MTBC genome in a relational database can be represented as a high-dimensional dataset for multiway analysis. The multiple-biomarker tensor is constructed this way, with one of the modes representing strains and other modes representing biomarkers. In our experiments, we use this multidimensional array or tensor with three modes representing strains, spoligotype deletions, and MIRU patterns. This multiple-biomarker tensor captures three key properties of MTBC strains: spoligotype deletions, number of repeats in MIRU loci, and coexistence of spoligotype deletions with MIRU loci.

We used the tensor clustering framework to cluster MTBC strains using multiple biomarkers, and compared the clustering to SpolDB4 sublineages. Next, we used supervised tensor learning and classified MTBC strains into major lineages using spoligoype deletions and MIRU patterns. We compared multiway and two-way supervised learning methods based on their prediction accuracy for major lineage

Table 3.3: Number of components used in PARAFAC and Tucker3 models to fit the tensors for the datasets to be clustered. We used the core consistency diagnostic, denoted as CC in the table, to validate PARAFAC models and percentage of explained variance to validate Tucker3 models.

Major Lineage	Tensor size	PARAFAC		Tucker3	
		# Components	CC / Variance	# Components	Variance
M. africanum	64 × 22 × 12	3	95.08 / 93.33	[4 3 1]	91.94
M. bovis	102 × 34 × 12	2	100.00 / 86.02	[7 5 1]	91.05
East Asian (Beijing)	571 × 5 × 12	2	100.00 / 81.58	[3 4 2]	93.09
East-African Indian (CAS)	508 × 18 × 12	3	90.75 / 80.48	[6 6 4]	94.27
Indo-Oceanic	1023 × 28 × 12	5	92.99 / 80.35	[15 13 5]	95.55
Euro-American	4580 × 109 × 12	14	99.06 / 89.83	[14 13 5]	89.77

classification. In the following section, we use the unsupervised tensor clustering framework on multiple-biomarker tensors to subdivide major lineages of MTBC into sublineages.

3.3.1 Subdivision of major lineages into sublineages

We subdivide each major lineage of MTBC into sublineages using multiple-biomarker tensors. For each major lineage, we generated the multiple-biomarker tensor using spoligotypes and MIRU types and applied multiway models to identify putative sublineages of each major lineage. Two multiway analysis methods were used: PARAFAC and Tucker3. Details of the methods and how the model parameters or components were selected can be found in the methods section. The validated multiway models with numbers of components for each major lineage are shown in Table 3.3. To evaluate the resulting clusters, we compared them to the published SpolDB4 families for each major lineage. The results are summarized in Table 3.4. We used the F-measure to measure how well the tensor sublineages match the SpolDB4 families with 1 indicating an exact match and 0 indicating no match. The average best-match stability is used to assess certainty of tensor sublineages respectively with 1 indicating highly stable clusters. For each major lineage, results show that the tensor analysis finds highly stable sublineages (the best-match stability is ≥84%) and that the number of sublineages found using tensors is close but not always identical to the number of SpolDB4 families.

The F-measure values range from 53% to 88% indicating that the sublineages found by the tensors only partially overlap with those of SpolDB4. Recall that the

Table 3.4: F-measure and average best-match stability are used to assess the agreement of the tensor sublineages to the SpolDB4 lineages and certainty of tensor sublineages respectively.

Major Lineage	# SpolDB4 families	# Tensor sublineages	F-measure	Stability
M. africanum	4	4	0.66	1
M. bovis	5	3	0.71	1
East Asian (Beijing)	2	6	0.88	1
East-African Indian (CAS)	4	4	0.75	1
Indo-Oceanic	13	9	0.67	0.86
Euro-American	33	35	0.53	0.84

SpolDB4 families were created by expert analysis using only spoligotypes and that analysis by alternative biomarkers such as SNP and LSP has led to alternative definitions of MTBC sublineages. The tensor sublineages are based on spoligotype and MIRU patterns, thus in some cases the tensor divides SpolDB4 families due to difference in MIRU patterns even if the spoligotypes match. In other cases, the tensor analysis merges the SpolDB4 families because the collective spoligotypes and MIRU patterns are very close. In some cases, the tensor analysis almost exactly reproduces a SpolDB4 family providing strong support for the existence of these families with no expert guidance. In addition, the MIRU patterns provide additional evidence for the existence of these distinct sublineages. Thus, multiway analysis of MTBC strains of each major lineage with multiple biomarkers leads to new sublineages and reaffirms existing ones. Further insight can be obtained by examining the putative sublineages for each major lineage, which is detailed next.

3.3.1.1 *M. africanum*

The most stable clusters were produced using PARAFAC and it constructed four putative sublineages of *M. africanum*, denoted MA1 to MA4. Table 3.5 gives the stability of each sublineage and the correspondence between the tensor sublineages and the SpolDB4 families. These four putative sublineages are quite distinct as shown by the stability of 1 for each sublineage and the clear separation of the four sublineages in the PCA plot in Figure 3.6. Figure 3.7 shows heat maps representing the spoligotype and MIRU signatures for each tensor sublineage, with white indicating 0 probability and black indicating probability of 1.

The tensor sublineages strongly support the existence of the SpolDB4 AFRI_1, AFRI_2 and AFRI_3 families and show that the AFRI family is composed of these

Table 3.5: Confusion matrix for 64 distinct *M. africanum* strains showing the correspondence between the SpolDB4 families and tensor sublineages. The stability of each tensor sublineage is given in the second row. All four *M. africanum* sublineages have a stability of 1, indicating that clear and distinct genetic diversity exists between the *M. africanum* sublineages. Each number in the table represents the number of strains that belong to associated SpolDB4 lineage in that row and associated tensor sublineage in that column.

	MA1	MA2	MA3	MA4
Stability	1	1	1	1
AFRI	2	5	1	0
AFRI_1	21	0	0	16
AFRI_2	0	12	0	0
AFRI_3	0	1	6	0

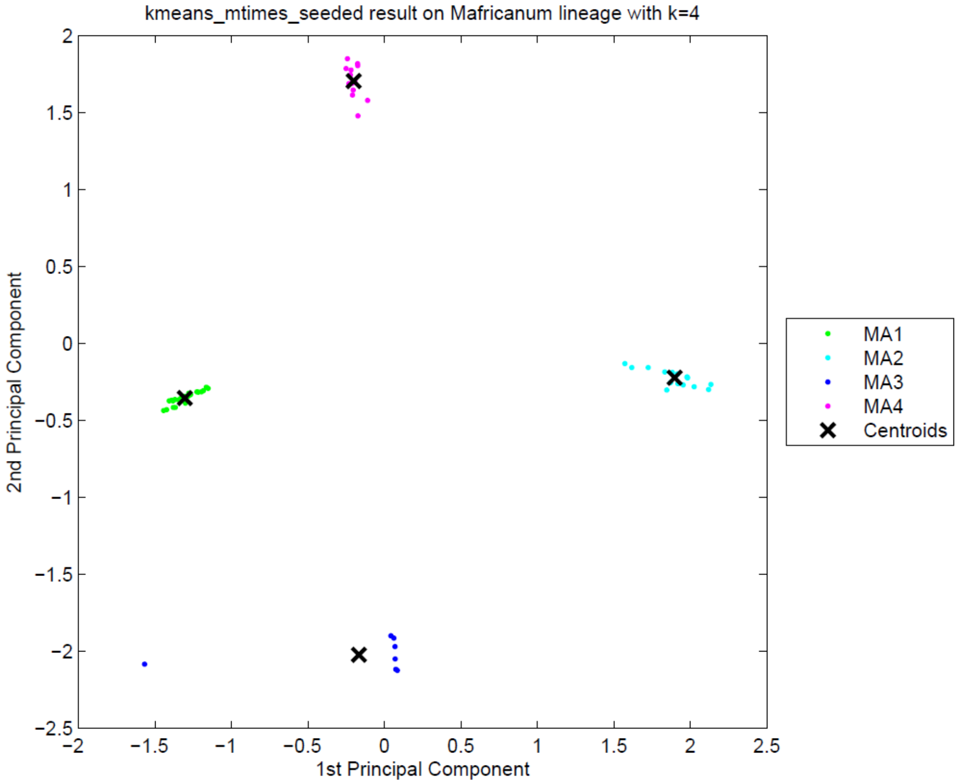


Figure 3.6: Clustering plot of *M. africanum* strains using Principal Component Analysis on the score matrix obtained from the PARAFAC model. Four putative tensor sublineages, MA1 to MA4, are clearly distinct along the principal component axes.

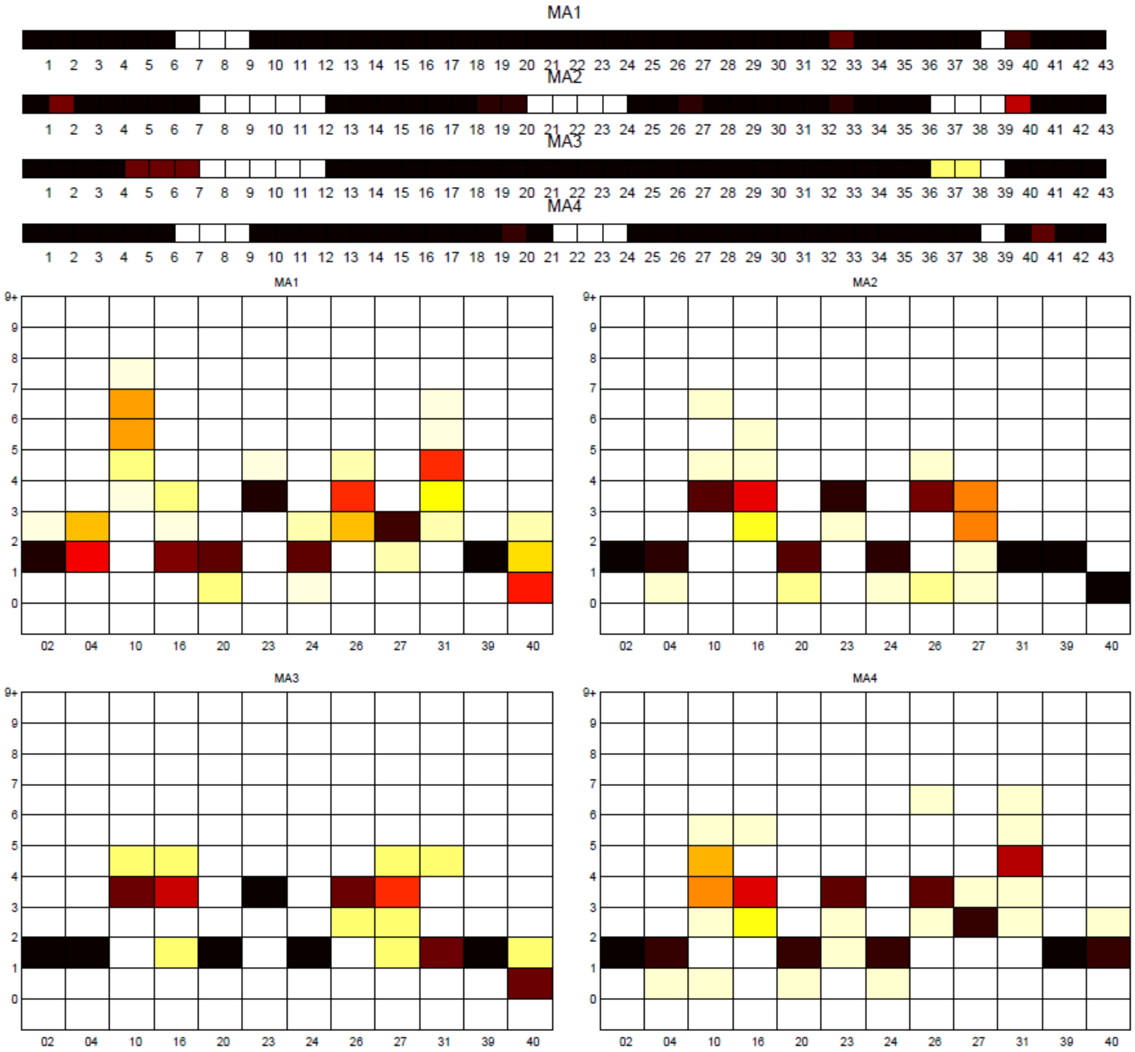


Figure 3.7: Spoligotype and MIRU signatures of tensor sublineages of *M. africanum* strains. White indicates probability of 0 and black indicates probability of 1. Intermediate colors represent probabilities in the range (0, 1). MA1 and MA4 are similar in their MIRU signatures, and MA4 strains lack spacers 22 through 24, in addition to the deletions of MA1 strains. MIRU signatures of MA2 and MA3 strains are also similar, and MA2 has an extra deletion, 21 through 24, in addition to the deletions of MA3 strains.

Table 3.6: Confusion matrix for 64 distinct *M. africanum* strains showing the correspondence between the West African 1 and 2 sublineages and tensor sublineages. For the data not from MIRU-VNTR*plus*, the lineage is indicated as unspecified.

	MA1	MA2	MA3	MA4
West African 1	0	5	0	0
West African 2	21	0	0	16
Unspecified	2	13	7	0

three families. With an F-measure of 66%, the tensor sublineages differ markedly from the SpolDB4 families for the *M. africanum* lineage. The AFRI family results largely explain this difference – AFRI is spread across three tensor sublineages. Disregarding AFRI, sublineages MA2 and MA3 match families AFRI_2 and AFRI_3 respectively. Interestingly, AFRI_1 is further subdivided into sublineages MA1 and MA4. The spoligotypes in MA1 and MA4 differ by only one contiguous deletion of spacers 22 through 24, but their MIRU signatures clearly distinguish them especially in MIRU loci 10, 16 and 40. The tensor indicates that the AFRI sublineage classification defines somewhat generic *M. africanum* strains that can be distinctly placed in the groups MA1 (part of AFRI_1), MA4 (other part of AFRI_1), MA2 (AFRI_2) and MA3 (AFRI_3).

The MIRU-VNTR*plus* labels, determined on the basis of LSPs, indicate that there are two sublineages, West African 1 and West African 2, within *M. africanum*. Table 3.6 indicates the correspondence between the tensor sublineages and MIRU-VNTR*plus* labels. MA1 and MA4 correspond to West African 2 and MA2 corresponds to West African 1. There is no data labeled by MIRU-VNTR*plus* in MA3, but we speculate that it is West African 1 since MA2 and MA3 have more closely related MIRU and spoligotype signatures.

3.3.1.2 *M. bovis*

PARAFAC generated the most stable clusters and constructed 3 sublineages for *M. bovis*, MB1, MB2, and MB3, while the dataset contains 5 SpolDB4 families, BOV, BOVIS1, BOVIS1_BCG, BOVIS2, and BOVIS3. Table 3.7 gives the correspondence between the tensor sublineages and the SpolDB4 families. All clusters

Table 3.7: Confusion matrix of *M. bovis* strains clustered into 3 groups using PARAFAC. Correct labels are SpolDB4 labels on the rows, and tensor sublineages are represented by each column. Stability of 1 for the tensor sublineages indicates that they have clean and marked differences based on their genotype. MB1 contains all BOVIS2 strains, MB2 contains all BOVIS3 strains, and MB3 contains all BOVIS1 and BOVIS1_BCG strains.

	MB1	MB2	MB3
Stability	1	1	1
BOV	7	5	5
BOVIS1	0	0	29
BOVIS1_BCG	0	0	11
BOVIS2	24	0	0
BOVIS3	0	21	0

have perfect stability and are well distinguished in the PCA plot in Figure 3.8. Figure 3.9 shows heat maps representing the spoligotype and MIRU type signatures of tensor sublineages. Much like the *M. africanum* SpolDB4 AFRI family, the BOV family defines a generic *M. bovis* sublineage that spreads across all three tensor sublineages. Disregarding BOV, MB3 consists of all of BOVIS1 and BOVIS1_BCG strains. Since BOVIS1_BCG is the attenuated bacillus Calmette-Guérin (BCG) vaccine strain, it is difficult to distinguish it from BOVIS1 using only MIRU patterns and spoligotypes. Therefore, the merger of BOVIS1 and BOVIS1_BCG is expected given the genetic similarity between the two groups of strains. Disregarding BOV, the MB1 and MB2 sublineages exactly match the SpolDB4 families BOVIS2 and BOVIS3 respectively.

3.3.1.3 East Asian (Beijing)

The most stable clusters are produced by Tucker3 and it constructs six distinct sublineages of East Asian (Beijing), denoted B1 through B6. The variability in the spoligotypes of East Asian is limited to spacers 35 through 43 since all East Asian strains have spacers 1 to 34 absent. Since the SpolDB4 classification is based only on spoligotypes, the limited variability allows only two families, BEIJING and BEIJING-LIKE. Table 3.8 shows the correspondence between tensor sublineages and

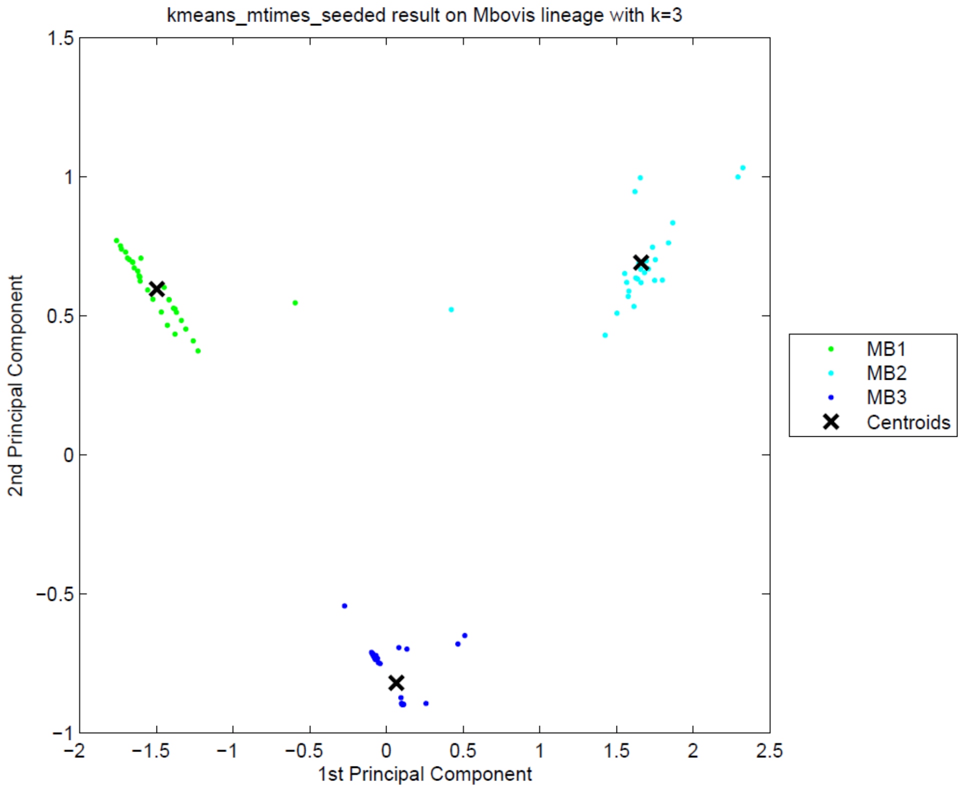


Figure 3.8: Clustering plot of *M. bovis* strains using Principal Component Analysis. Three putative tensor sublineages, MB1 to MB3, are clearly separated.

the SpolDB4 families. The clustering plot of tensor sublineages is shown in Figure 3.10. Heat maps representing the spoligotype and MIRU type signatures of tensor sublineages are shown in Figure 3.11. The tensor cleanly subdivides BEIJING into three sublineages B1, B4 and B6, all with stability 1. Spoligotype signatures of these sublineages differ. B1 strains have spacers 35 through 43 present, whereas B4 strains lack spacer 37, and B6 strains lack spacer 40. MIRU signature of sublineage B4 is clearly distinct in MIRU locus 40, having 3 repeats for most strains. The tensor subdivides the BEIJING-LIKE into sublineages B2, B3 and B5, each with distinct spoligotype signature. They all lack spacers 35 through 36. In addition, B2 strains lack spacer 37, and B3 strains lack spacer 40. Thus, the tensor strongly supports the existence of BEIJING and BEIJING-LIKE families, but also suggests that they can be further subdivided.

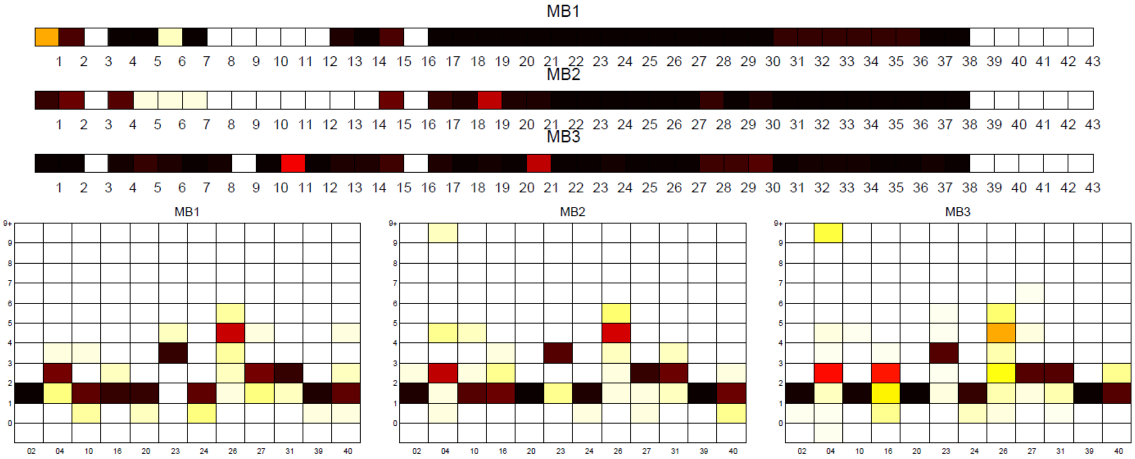


Figure 3.9: Spoligotype and MIRU signatures of tensor sublineages of *M. bovis* strains. Although MIRU signatures of MB1 and MB2 strains are similar, spoligotype signatures of MB1 and MB2 strains are clearly distinguishable by extra deletions of 13 through 14 in all MB2 strains, and deletions of 5 through 7 in some MB2 strains.

Table 3.8: Confusion matrix of East Asian (Beijing) strains clustered into 6 groups using Tucker3. Correct labels are SpolDB4 labels on the rows, and tensor sublineages are represented by each column. The six highly stable tensor sublineages are indicative of additional genetic diversity within the BEIJING and BEIJING-LIKE sublineages.

	B1	B2	B3	B4	B5	B6
Stability	1	1	1	1	1	1
BEIJING	468	0	0	18	0	41
BEIJING-LIKE	0	16	8	0	20	0

3.3.1.4 East-African Indian (CAS)

Tucker3 generated the most stable clusters and it constructed four distinct sublineages for East-African Indian (also known as CAS) denoted C1, C2, C3, and C4. The strains are also labeled with four SpolDB4 lineages: CAS, CAS1_DELHI, CAS1_KILI and CAS2. Table 3.9 shows the correspondence of tensor sublineages and SpolDB4 families. Figure 3.12 shows the clustering plot of tensor sublineages and Figure 3.13 shows spoligotype and MIRU type signatures of tensor sublineages.

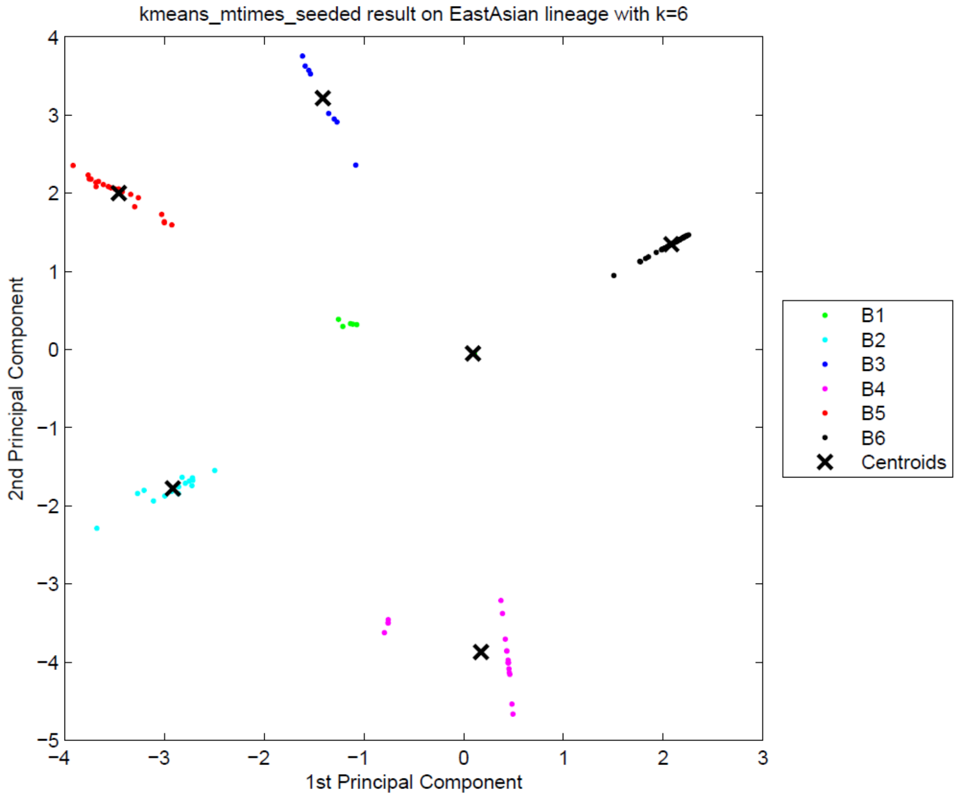


Figure 3.10: Clustering plot of East Asian (Beijing) strains using Principal Component Analysis. Six putative tensor sublineages, B1 to B6, are clearly distinct.

All sublineages are highly stable with stability 1. Much like with AFRI and BOV, the generic CAS family is divided across all tensor sublineages. C3 only contains CAS strains. Disregarding CAS, C1 contains most CAS1_DELHI strains and all CAS2 strains. C4 contains all CAS1_KILI strains. C2 contains 2 CAS1_DELHI strains, but the vast majority (331 strains) of CAS1_DELHI strains fall in C1. In addition to the common deletions of East-African Indian (CAS) strains, C2 strains lack spacer 22, C3 strains lack spacers 20 through 22, and C4 strains lack spacers 20 through 22 and spacer 35. Variabilities in MIRU loci 10, 26, 31 and 40 are also key to defining differences in the sublineages. C2 and C3 strains differ by variations in MIRU locus 10. C4 strains which include all CAS1_KILI strains exhibit a very distinct MIRU signature compared to other tensor sublineages, especially in MIRU locus 26.

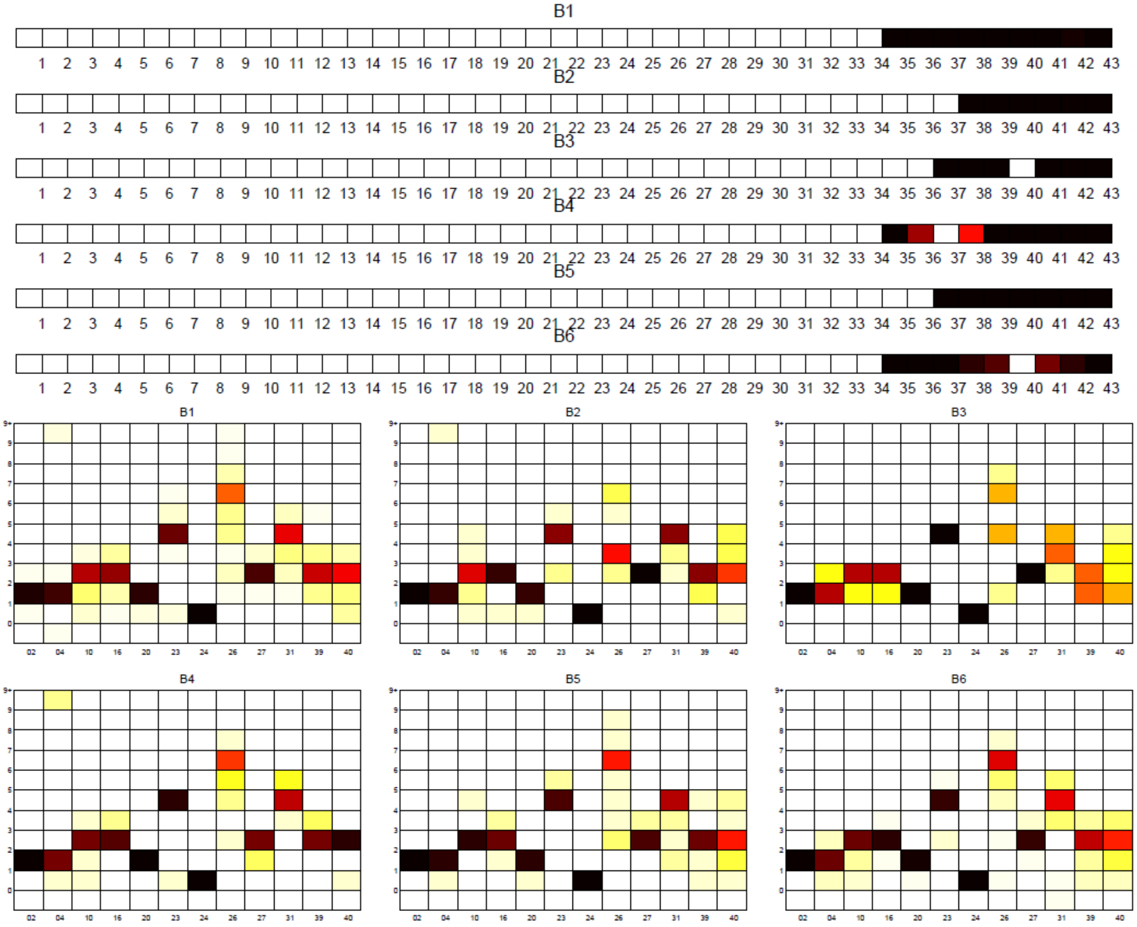


Figure 3.11: Spoligotype and MIRU signatures of tensor sublineages of East Asian (Beijing) strains. Tensor sublineages B1, B4, B6 include BEIJING strains and sublineages B2, B3, B5 include BEIJING-LIKE strains.

Table 3.9: Confusion matrix of East-African Indian (CAS) strains clustered into 4 groups using Tucker3. Correct labels are SpolDB4 labels on the rows, and tensor sublineages are represented by each column.

	C1	C2	C3	C4
Stability	1	1	1	1
CAS	50	21	35	1
CAS1_DELHI	331	2	0	0
CAS1_KILI	0	0	0	23
CAS2	45	0	0	0

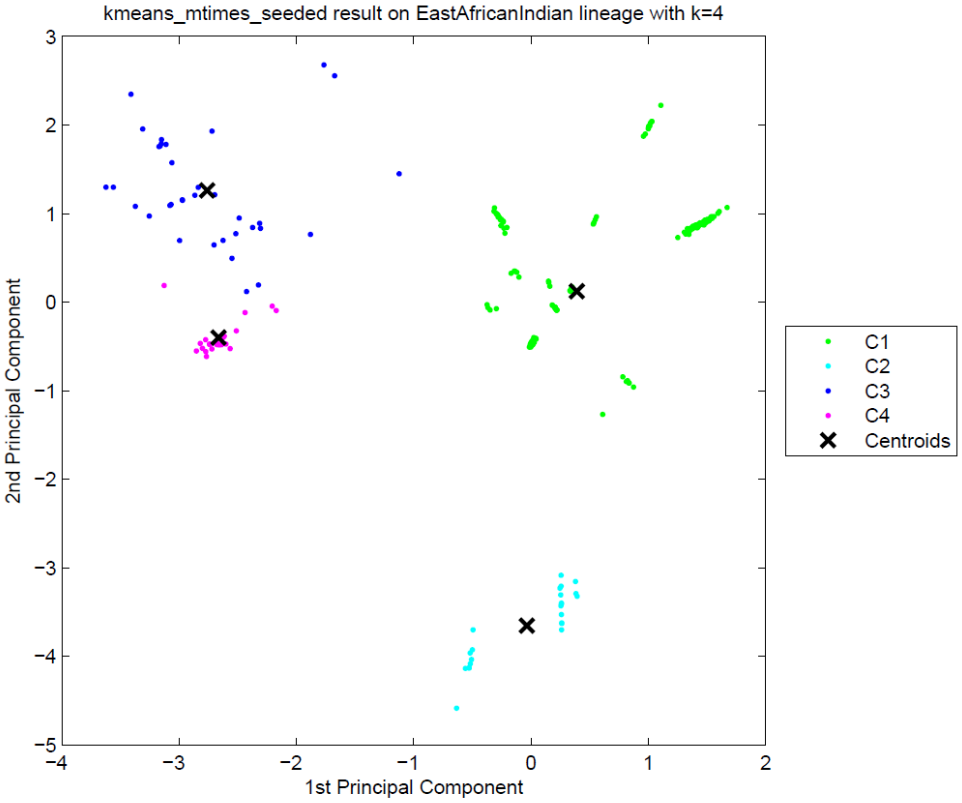


Figure 3.12: Clustering plot of East-African Indian (CAS) strains using Principal Component Analysis. Four putative tensor sublineages, C1 to C4, are clearly distinct.

3.3.1.5 Indo-Oceanic

PARAFAC found the most stable clusters and it constructs nine distinct putative sublineages for Indo-Oceanic, denoted IO1 to IO9, while the dataset has thirteen SpolDB4 lineages. Table 3.10 shows the correspondence of tensor sublineages and SpolDB4 families. Figure 3.14 shows the clustering plot of tensor sublineages and Figure 3.15 shows spoligotype and MIRU signatures of tensor sublineages. The EAI5 family acts much like the CAS, BOV, and AFRI families, spreading across all the Indo-Oceanic sublineages except IO4. The small MANU1 family also spreads across four sublineages. The existence of the MANU1 family has not been well established by other biomarkers. Disregarding these two troubling families, the tensor sublineages correspond closely to the SpolDB4 families. Table 3.10 shows that there is almost a one-to-one mapping between most SpolDB4 lineages and Indo-Oceanic tensor sublineages. Specifically, the mapping between the most stable clusters (with sub-

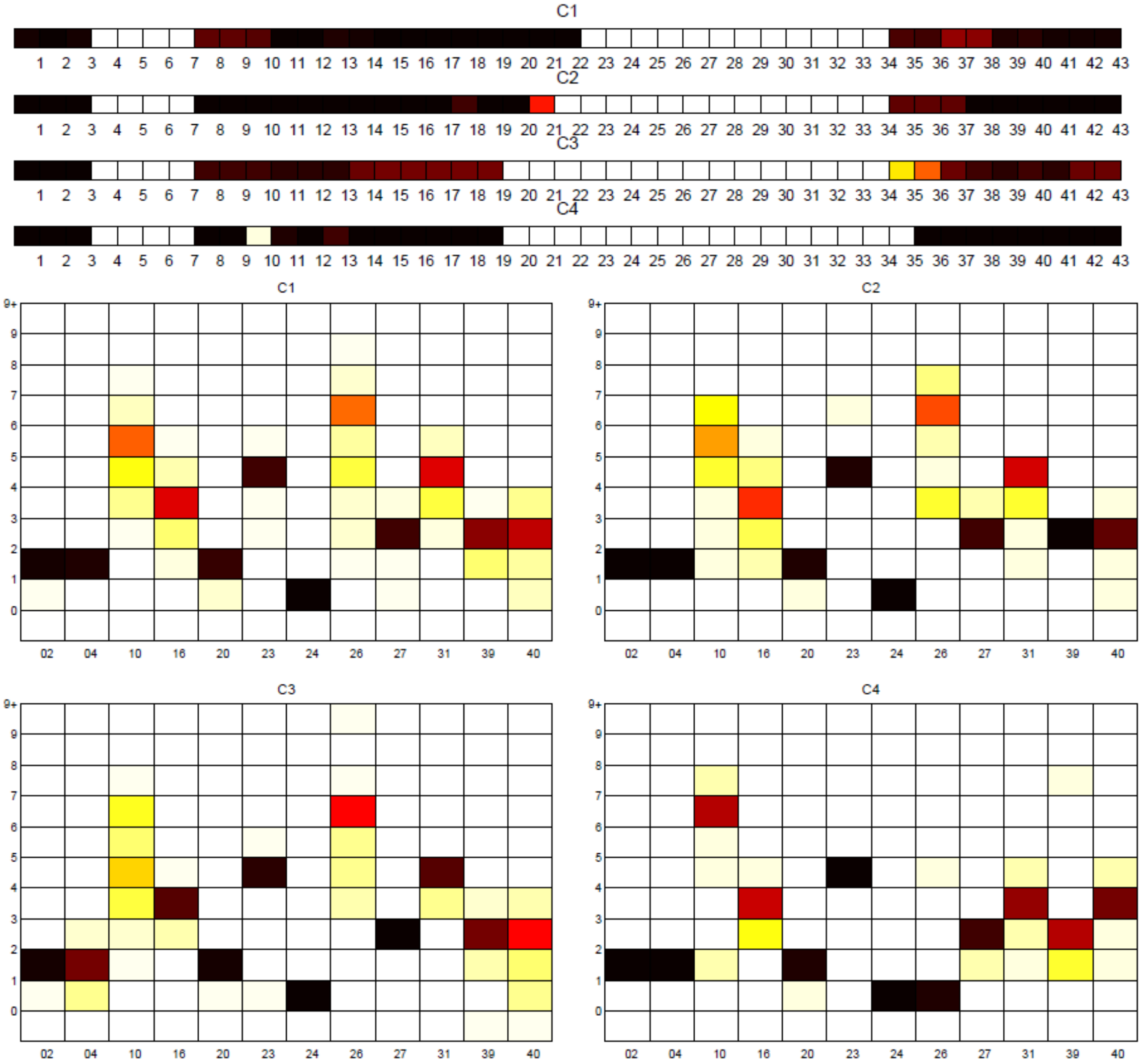


Figure 3.13: Spoligotype and MIRU signatures of tensor sublineages of East-African Indian (CAS) strains. In addition to deletions in C1 strains, C2 strains lack spacer 22. In addition to deletions in C3 strains, C4 strains lack spacer 35 and have only 1 repeat in MIRU 26. C2 and C3 strains are very close in their MIRU signature, but they differ by variations in MIRU locus 10.

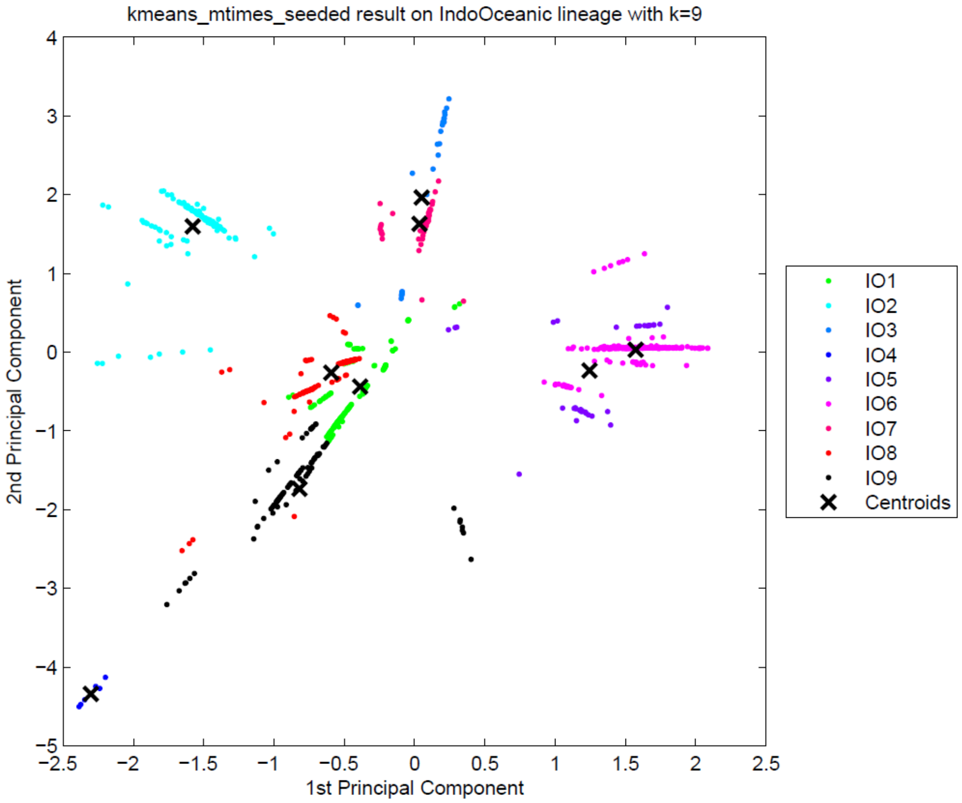


Figure 3.14: Clustering plot of Indo-Oceanic strains labeled by putative tensor sublineages using Principal Component Analysis. The tensor sublineages are not as distinct as it was for the previously analyzed major lineages, implying that the tensor sublineages are well distinguished in the PCA plot if they are stable.

lineage stability) and the families are: IO1 (.94) equals EAI6_BDG1, IO2 (1) equals EAI3_IND, IO4 (1) equals ZERO, and IO6 (.91) equals most of EAI2_MANILLA. All EAI strains are in IO9 (.77), all EAI1 strains are in IO8 (.86), all MICROTI strains are in IO5 (0.56), and all ZERO strains are in IO4. All EAI2_NTB strains are in IO5, all EAI3_IND strains are in IO2, and all EAI8_MDG strains are in IO7 (.84). EAI2_MANILLA is divided into two sublineages: 11 strains in IO5, 265 strains in IO6. While the spoligotype and MIRU signatures show that there are distinct EAI5 subgroups, the definition of the EAI5 and MANU1 groups are not well supported by the tensor analysis. They may represent a more generic sublineage that is further subdivided. Distinct patterns are observable in the spoligotype and MIRU signatures for most of the tensor sublineages.

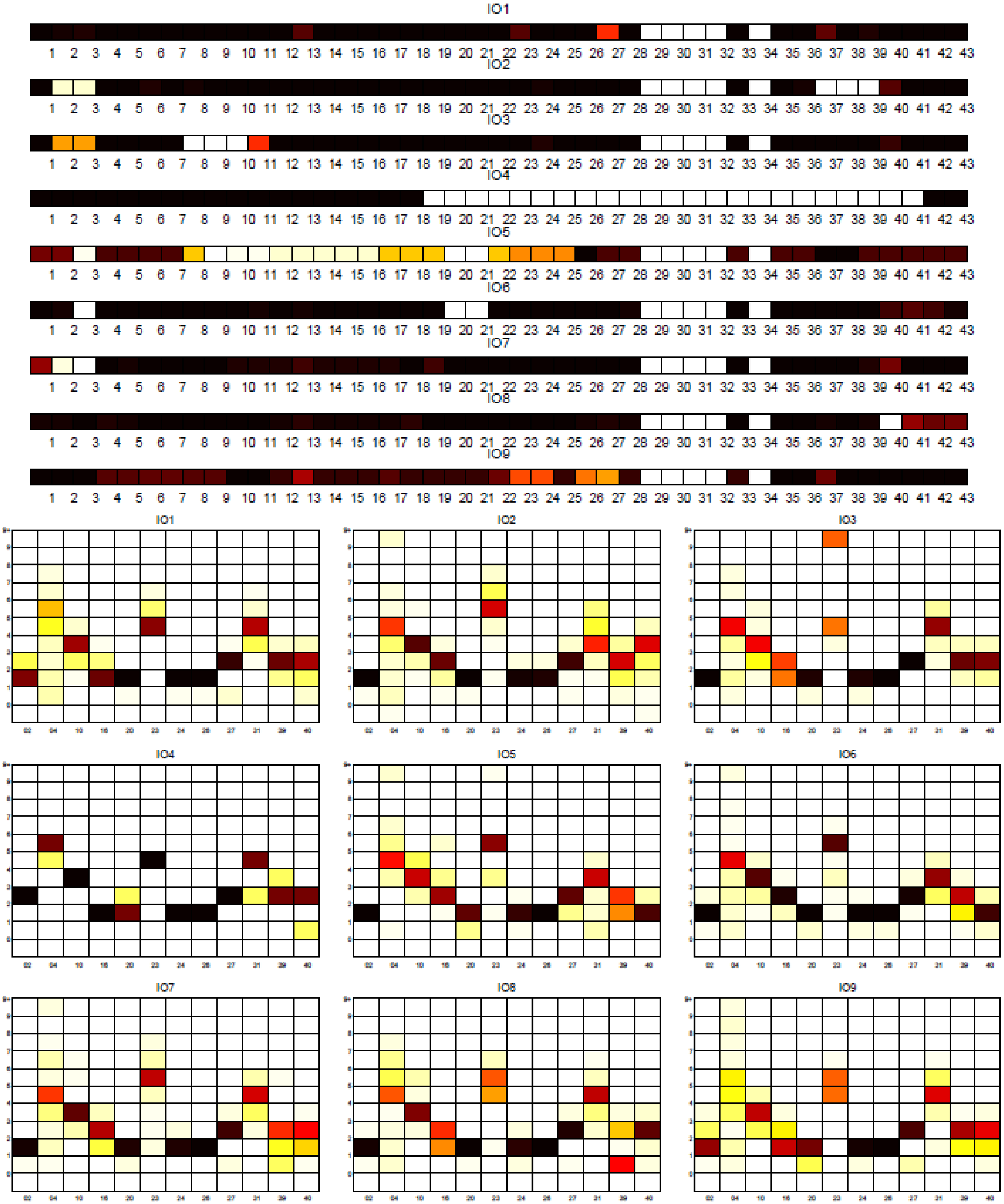


Figure 3.15: Spoligotype and MIRU signatures of tensor sublineages of Indo-Oceanic strains.

Table 3.10: **Confusion matrix of Indo-Oceanic strains clustered into 9 groups using PARAFAC. Correct labels are SpolDB4 labels on the rows, and tensor sublineages are represented by each column. SpolDB4 lineages except EAI5 and MANU1 map to distinct tensor sublineages.**

	IO1	IO2	IO3	IO4	IO5	IO6	IO7	IO8	IO9
Stability	0.94	1	0.90	1	0.56	0.91	0.84	0.86	0.77
EAI	0	0	0	0	0	0	0	0	6
EAI1	0	0	0	0	0	0	0	2	0
EAI1_SOM	0	0	2	0	0	0	8	107	0
EAI2_MANILLA	0	0	0	0	11	265	0	0	0
EAI2_NTB	0	0	0	0	15	0	0	0	0
EAI3_IND	0	105	0	0	0	0	0	0	0
EAI4_VNM	0	0	0	0	0	0	0	3	42
EAI5	231	24	26	0	3	10	35	32	31
EAI6_BGD1	33	0	0	0	0	0	0	0	10
EAI8_MDG	0	0	0	0	0	0	4	0	0
MANU1	1	0	0	0	0	5	0	2	1
MICROTI	0	0	0	0	3	0	0	0	0
ZERO	0	0	0	6	0	0	0	0	0

3.3.1.6 Euro-American

Tucker3 found the most stable clusters and it generates 35 sublineages for Euro-American, denoted E1 to E35, while there are 33 SpolDB4 lineages labeled Euro-American. See additional file at http://www.biomedcentral.com/1471-2164/12/S2/S1/additional/ for the confusion matrix of Euro-American strains that shows the correspondence of tensor sublineages and SpolDB4 families. Figure 3.16 shows the clustering plot of tensor sublineages. Figure 3.17 and Figure 3.18 show the spoligotype and MIRU signatures of tensor sublineages respectively.

Strains belonging to families H2, H37Rv, LAM12_MAD1, T1 (Tuscany variant), T1_RUS2, T4, T5_MAD2, and T5_RUS1 are clustered in tensor sublineages E9, E7, E8, E24, E11, E34, E34, and E17 respectively. In contrast, the T1 family, an ancestor strain family, is distributed across 25 tensor sublineages, with most T1 strains in E34. Sublineage stability is above .90 for 18 tensor sublineages. Spoligotype and MIRU signatures of sublineages suggest either subdivision or merging of SpolDB4 families. For instance, tensor sublineages E2, E6, and E32 include T1

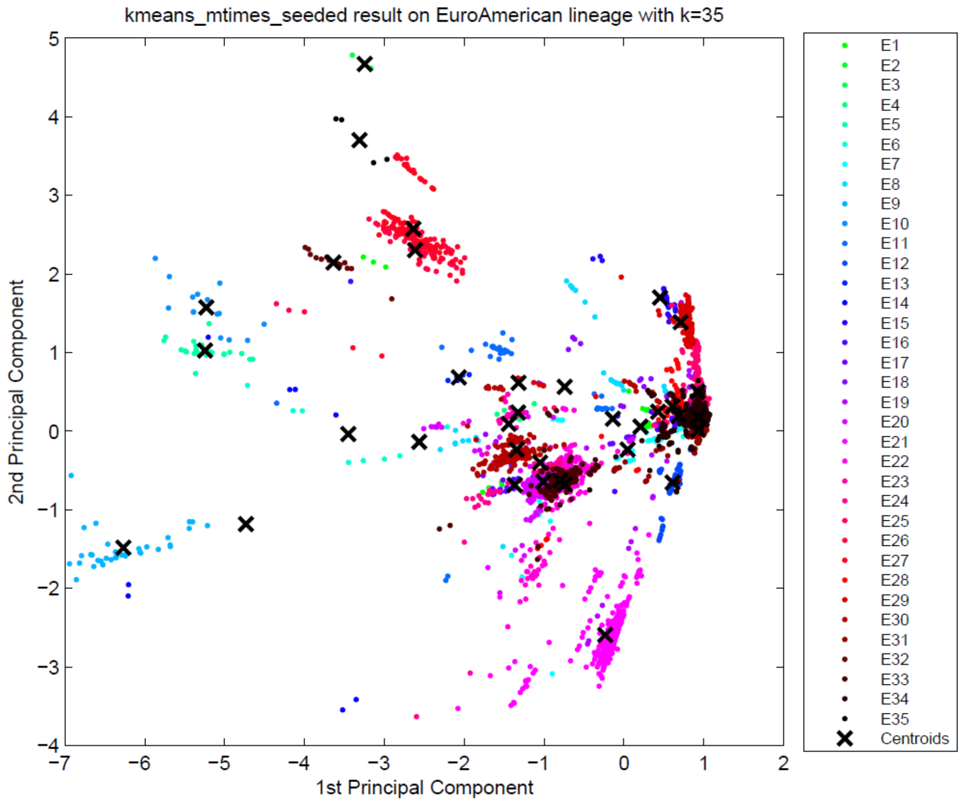


Figure 3.16: Clustering plot of Euro-American strains labeled by 35 tensor sublineages using Principal Component Analysis. The tensor sublineages are not as distinct as they were for the previously analyzed major lineages, reflecting the variability in the tensor cluster stability. It may also be due to the anticipated hierarchical structure in Euro-American strains.

strains only. In addition to common spacer deletions of Euro-American strains, E2 strains lack spacers 15 through 26, E6 strains lack spacers 9 through 23, and E32 strains lack spacers 1 through 19, which are all variations in spoligotype signatures of T1 strains. This sublineage classification further subdivides the poorly-defined ancestor T1 family. Strains of LAM families on the other hand are grouped in 17 tensor sublineages. Prior studies have found that LAM Rio strains identified by SNPs are found in multiple SpolDB4 lineages [121]. Therefore, it is expected that the use of multiple biomarkers leads to subdivision or merging of some SpolDB4 families.

Although most stable clusters of the Euro-American strain dataset are found using best-match stability, the DD-weighted gap statistic plot has multiple peaks.

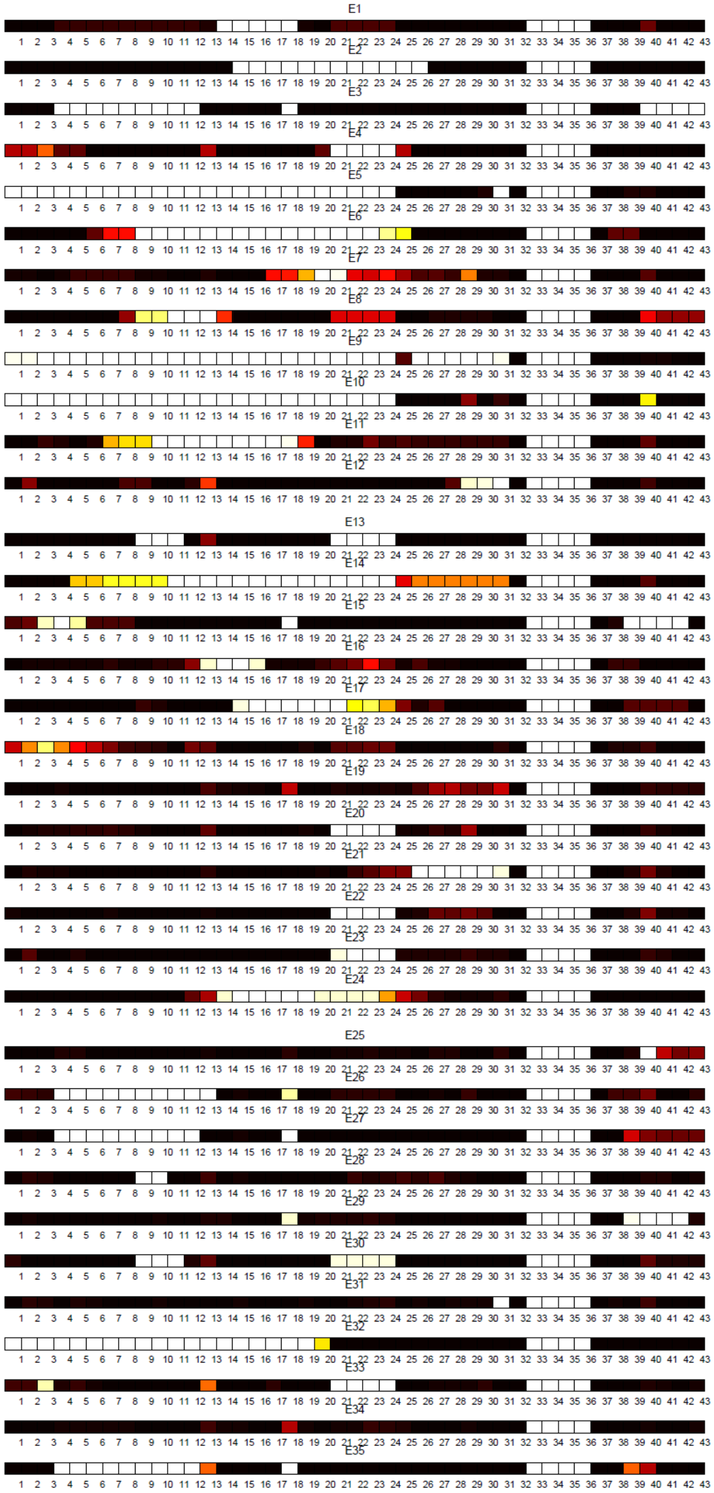


Figure 3.17: Spoligotype signatures of tensor sublineages of Euro-American strains.

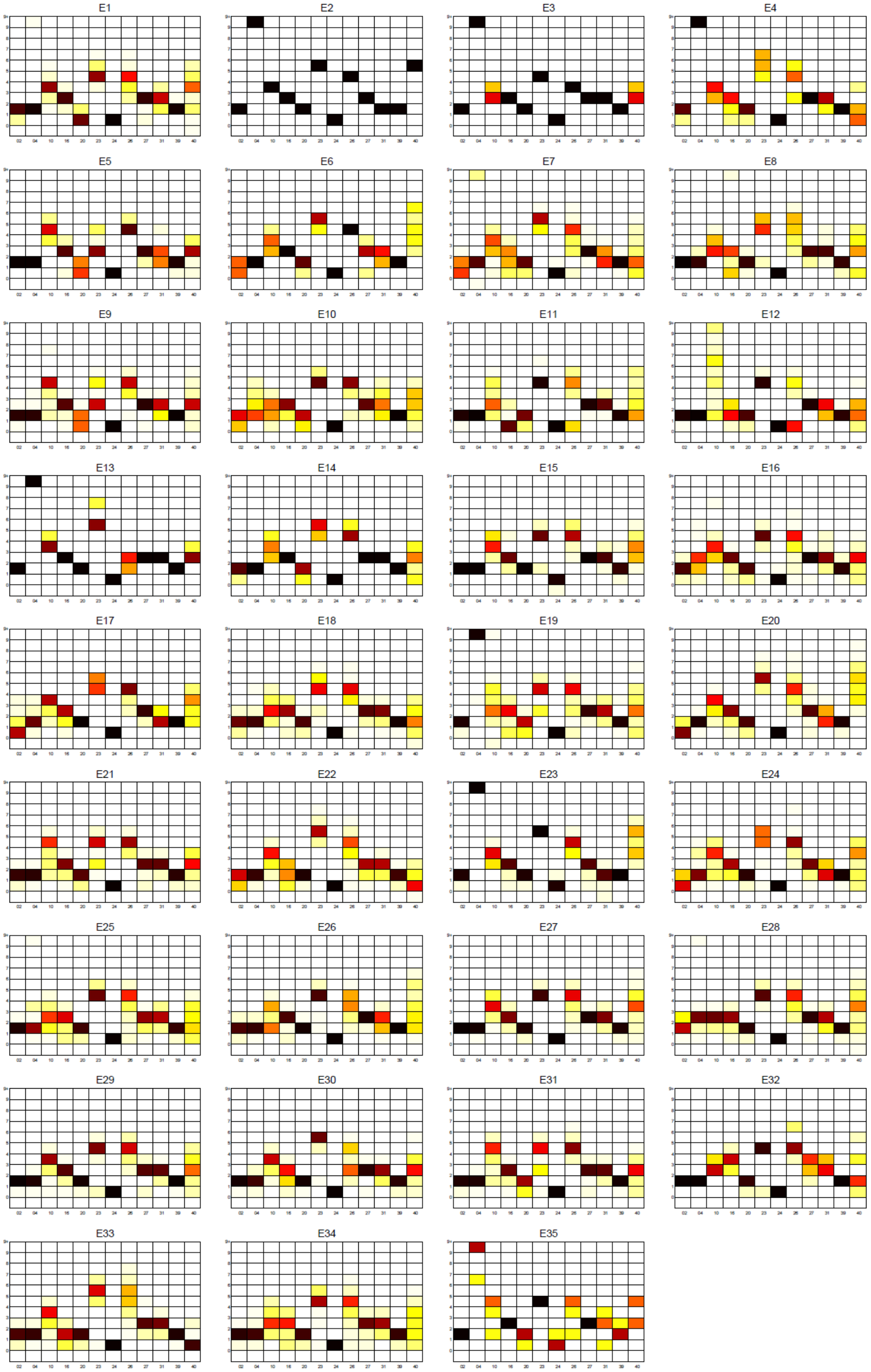


Figure 3.18: MIRU signatures of tensor sublineages of Euro-American strains.

DD-weighted gap statistic, detailed in the methods section, is a cluster validity measure which is also used for detecting hierarchical structure in the datasets. Multiple peaks in DD-weighted gap statistic plot suggest that the Euro-American dataset may have a multi-level hierarchical structure. Model order selection with randomized maps by Bertoni and Valentini can be used to detect the hierarchical structure in the Euro-American dataset [122].

We used the unsupervised tensor clustering framework to cluster MTBC strains of major lineages into sublineages. Next, we turn our attention to supervised tensor learning methods on multiple-biomarker tensors to classify strains into major lineages.

3.3.2 Classification of MTBC strains into major lineages using two-way and multiway supervised learning

Multiple-biomarker tensors can be used in supervised classification models as well as in unsupervised models. We use multiway partial least squares (N-PLS) on multiple-biomarker tensors to predict major MTBC lineages [116]. In our experiments, we used spoligotype and MIRU as biomarkers and predicted the six major lineages using the same data as for the above unsupervised learning experiments combined into a single dataset. More specifically, we used 12 spoligotype deletions found informative in major lineage classification combined with 12-loci MIRU [26]. We predicted major lineages with the N-PLS multiway method and compared it with standard two-way PLS and prior results for conformal Bayesian Networks [20]. Table 9 shows the average testing F-measure as estimated by 5-fold cross-validation. We generate the multiple-biomarker tensor using 12 spoligotype deletions and 12-loci MIRU with one additional bit indicating whether the at least one MIRU pattern includes letter rather than number of repeats, and create a predictive model using the N-PLS multiway method. The model for standard 2-way PLS is created by representing the data as a matrix with columns corresponding to 12 spoligotype deletions and 12-loci MIRU with the additional indicator bit, and rows corresponding to MTBC strains. The number of latent variables for both N-PLS and PLS are selected by inner 4-fold cross-validation of the training set data only.

Table 3.11: Multiway N-PLS and standard two-way PLS classification accuracy results when 12 spoligotype deletions and MIRU patterns are used to classify MTBC strains into major lineages. Highly accurate classification results compare favorable to prior results based on a conformal Bayesian Network in [20].

Method	Average F-measure
N-PLS	0.9961 ± 0.0009
Standard PLS	0.9955 ± 0.0017
Conformal Bayes Net	0.9897

We compare N-PLS, standard PLS and Conformal Bayes Network (CBN) methods by F-measure of major lineage classification and see that they are accurate predictive models with no significant difference between the approaches. Table 3.11 shows the F-measure values for N-PLS, standard PLS and CBN. The average F-measure of major lineage prediction on the same data using the CBN is 0.9897 [20]. This shows that N-PLS and standard PLS methods predict major lineages as accurately as CBN, with a slightly better average F-measure value. All three methods achieve outstanding results for major lineage classification with no significant difference between approaches.

3.4 Conclusion

This study investigates multiple-biomarker tensors and illustrates how they can be used for both unsupervised and supervised learning models. First, a novel clustering framework is used to analyze the sublineage structure of MTBC strains based on multiple biomarkers. We generated multiple-biomarker tensors to represent multiple biomarkers of the MTBC genome and used multiway models for dimensionality reduction. The multiway representation determines a transformation of the data that captures similarities and differences between strains based on two distinct biomarkers. We clustered MTBC strains based on the transformed data using improved k-means clustering and validated clustering results. We evaluated the sublineage structure of major lineages of MTBC and found similarities and clear distinctions in our subdivision of major lineages compared to the SpolDB4 classification. Simultaneous analysis of spoligotype and MIRU through multiple-biomarker tensors and clustering of MTBC strains leads to coherent sublineages within major lineages with clear and distinctive spoligotype and MIRU signatures. Second, we demonstrated how the multiple-biomarker tensor can be used to predict major lineages with extremely high accuracy competitive with other approaches. We show that 3-way PLS, 2-way PLS and CBN models are accurate major lineage predictors for MTBC strains.

The tensor clustering framework is flexible and can be applied to any multidimensional strain data. The design of the resulting tensor depends on the question to be answered. In this study, multiple-biomarker tensors are designed to find groups of MTBC strains. Thus, the application of the tensor clustering framework on multiple-biomarker tensors leads to sublineages of MTBC within major lineages. The multiple-biomarker tensor is further validated by the fact that it can used to predict known major lineages with high accuracy using N-PLS. N-PLS with multiple-biomarker tensors can be used for semi-supervised learning as well. This can be useful for learning predictive models for sublineages in which only part of the data is labeled with sublineages and the other part of the data has no labels. This may result in more reliable and accurate classifiers of MTBC sublineages, and the resulting sublineage classifiers would be a significant enhancement to TB control, epidemiology and research. We leave this to future work.

The tensor clustering framework used in this study can be further extended to find subgroups of MTBC strains based on other biomarkers such as RFLP and SNPs. 15-loci MIRU and 24-loci MIRU patterns can also be used to represent MTBC genomes with multiple-biomarker tensors. Moreover, more than two biomarkers can be used in the MTBC genome representation. But, ambiguity in the tensor entries is an open question that needs to be solved in the tensor representation when more than two biomarkers are used. Addition of new biomarkers will increase the number of modes of the multiple-biomarker tensor, but the multiway analysis methods will remain the same.

Other questions of interest can be addressed by designing and analyzing host-pathogen tensors to examine the relationship of the pathogen genotype with host (or equivalent) attributes to examine questions of interest. For example, since the

MTBC sublineages are known to be highly geographically dependent, a tensor which combines the pathogen genotype with the country of birth of the host may reveal additional sublineage structure and transmission patterns. A tensor combining MTBC genotype and host disease phenotype such as site of infection and drug resistance could be used to analyze MTBC genotype-phenotype relations.

CHAPTER 4
INFERRED SPOLIGOFOREST TOPOLOGY UNRAVELS SPATIALLY BIMODAL DISTRIBUTION OF MUTATIONS IN THE DR REGION

4.1 Introduction

Tuberculosis (TB) is a leading cause of death among infectious diseases. Tuberculosis is caused by *Mycobacterium tuberculosis* complex (MTBC). One third of the human population is infected, either latently or actively, with TB [1]. DNA fingerprinting of MTBC strains is used for tracking the transmission of tuberculosis. Isolates from TB patients are genotyped using multiple biomarkers, which include spacer oligonucleotide types (spoligotypes), Mycobacterium Interspersed Repetitive Units - Variable Number Tandem Repeats (MIRU-VNTR), and IS*6110* Restriction Fragment Length Polymorphism (RFLP) [13, 14, 23].

Biomarkers of MTBC change over time. Brosch et al. presented an evolutionary repetition model based on the analysis of twenty regions of difference (RD) found in a comparison of whole genome sequences of MTBC clinical strains [11, 123]. Tanaka et al. introduced cluster-graphs to analyze genotype clusters of MTBC separated by a single mutation step [124]. Based on the observation that deletion length follows a Zipf distribution, Reyes et al. presented a probabilistic mutation model of spoligotypes to disambiguate the ancestors [89]. Grant et al. simulated stepwise loss or gain of repeats in MIRU loci using a stochastic continuous-time model, and suggested that all MIRU loci mutate very slowly [125].

In this study, we present a mutation model of spoligotypes based on variations

* Portions of this chapter previously appeared as:

C. Ozcaglar, A. Shabbeer, N. Kurepina, N. Rastogi, B. Yener, and K. P. Bennett, "Inferred spoligoforest topology unravels spatially bimodal distribution of mutations in the DR region," *IEEE Transactions on NanoBioscience*, in press, 2012.

C. Ozcaglar, A. Shabbeer, N. Kurepina, B. Yener, and K. P. Bennett, "Data-driven insights into deletions of *Mycobacterium tuberculosis* complex chromosomal DR Region using spoligoforests," in *Proc. 2011 IEEE Int. Conf. Bioinformatics and Biomedicine (BIBM)*, Atlanta, GA, pp. 75-82, 2011.

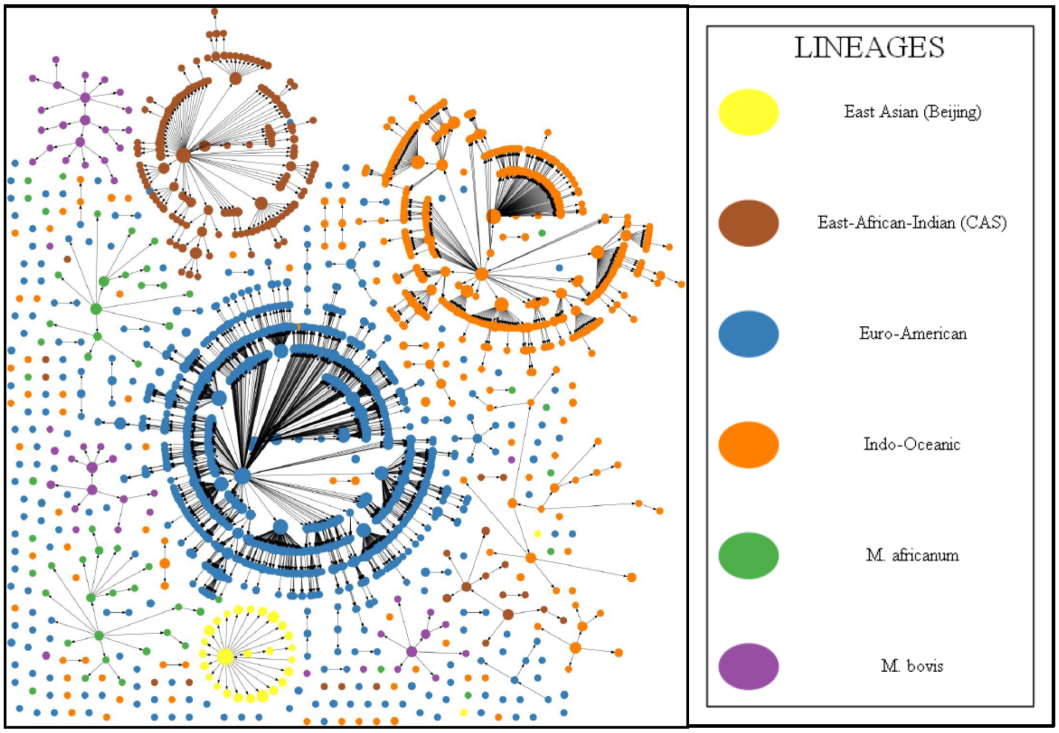


Figure 4.1: The spoligoforest of the CDC dataset. Each node represents a distinct spoligotype, and each edge represents a one-step mutation event from parent spoligotype to child spoligotype. Node sizes are proportional to the number of patients infected with MTBC strains having the spoligotype, in log scale. Nodes are colored by major lineages of MTBC strains. The spoligoforest generator is implemented in Java, using the visualization software Graphviz [126].

in the direct repeat (DR) region. To disambiguate the parents in the cluster-graph, we add an independent biomarker, MIRU-VNTR. First, we use a large patient dataset from the United States Centers for Disease Control and Prevention (CDC) and generate the most parsimonious forest of spoligotypes, called a spoligoforest. The spoligoforest generation is based on the contiguous deletion assumption, nonexistence of convergent evolution and three distance measures defined on spoligotypes and MIRU patterns. The spoligoforest of the CDC dataset in Figure 4.1 generated using this model displays the putative history of mutation events in the chromosomal DR region. Each node in the spoligoforest represents a distinct spoligotype, and each edge represents a potential mutation event from parent spoligotype to child spoligotype. The number of spacers lost in a mutation event is referred as the mutation length. We compare the DR evolution model to existing mutation models in terms of number of mutations and segregation accuracy and show that our mutation model with the additional biomarker, MIRU-VNTR, leads to as many within-lineage mutation events as in other mutation models. We identified topological attributes of the spoligoforest and gave insights into variations of spoligotypes. Based on the spoligoforest, the number of descendant spoligotypes follows a power law distribution. On the other hand, based on goodness-of-fit results, the mutation length frequency does not follow a power law distribution, in contrast to prior studies. The number of mutations at contiguous DR loci follows a bimodal distribution, and the modes are spacer 13 and spacer 40, which are hotspots, e.g. sites of increased observed variability. Spacer 34 is the change point in the distribution, and it is stable, which is due to lack of spacers 33-36 in most MTBC strains in the CDC dataset. We hypothesized that this bimodal distribution results in unobservable longer mutation events, which is why power law distribution is not a plausible fit to mutation length frequency. Based on this observation, we built two alternative models for mutation length frequency. The Starting Point Model (SPM) conditions the mutation length on the starting point of mutation, and the Longest Block Model (LBM) conditions the mutation length on the length of the longest contiguous block spacers beyond the starting point of mutation. Both SPM and LBM are plausibly good models for mutation length frequency distribution.

4.2 Background

In order to build a mutation model for evolution of the chromosomal DR region, we used two biomarkers of MTBC: spoligotypes and MIRU patterns. Each biomarker has a different mutation mechanism which is analyzed separately. Spoligotypes can lose spacers in the DR region, but not gain, while MIRU loci can either lose or gain tandem repeats [104,127,128]. In this section, we give a brief background on spoligotyping, MIRU-VNTR typing, and mutation of both biomarkers.

4.2.1 Spoligotyping

Spoligotyping is a PCR-based genotyping method of MTBC that exploits the polymorphism in the DR locus. The DR region belongs to the class of clustered regularly interspaced palindromic repeats (CRISPR) loci [129]. It comprises of directly repeating sequences of 36 bp, separated by unique spacer sequences of 36 to 41 bp [17]. One repeat sequence and the following spacer sequence together is termed a direct variable repeat (DVR). A spoligotype is composed of 43 spacers, which are represented by a 43-bit binary sequence, where zeros and ones indicate absence and presence of particular spacer in the DR locus respectively.

4.2.2 MIRU-VNTR typing

MIRU is a homologous 46-100 bp DNA sequence tandemly repeated and dispersed within intergenic regions of MTBC genome [130, 131]. Among 12-loci, 15-loci and 24-loci MIRU pattern analysis formats, we used 12-loci MIRU patterns in this study for genotyping MTBC [18]. The 12-loci MIRU pattern consists of loci 154 / MIRU02, 580 / MIRU04, 960 / MIRU10, 1644 / MIRU16, 2059 / MIRU20, 2531 / MIRU23, 2687 / MIRU24, 2996 / MIRU26, 3007 / MIRU27, 3192 / MIRU31, 4348 / MIRU39, and 802 / MIRU40. The MIRU pattern of an MTBC strain is represented as a vector of length 12, where each entry indicates the number of repeats in the specified MIRU locus.

4.2.3 Mutation of spoligotypes and repeats in MIRU loci

Spacers in the DR region can be lost as a result of chromosomal rearrangement event, but not gained [104, 128]. As a result, mutation of spoligotypes is unidirectional and can only result in variations of type 1→0 at each locus. Therefore, similar to Camin-Sokal parsimony, mutation of spoligotypes is irreversible [93, 132]. Moreover, in a one-step mutation event, one or more contiguous spacers can be deleted. We refer to the rule of irreversible mutation of contiguous spacers as the *contiguous deletion assumption* [104, 128].

Tandem repeats in a MIRU loci can either be lost or gained as a result of duplication or multiplication in a mutation event [127]. Therefore, mutations at

MIRU loci are bidirectional, and can result in increment or decrement in the number of repeats. The variations in number of repeats in MIRU loci are simulated using the stepwise mutation model [133, 134].

4.3 Methods

4.3.1 CDC dataset

The CDC dataset comprises of 9336 unique MTBC strains as determined by spoligotype and 12-loci MIRU patterns, collected by the United States Centers for Disease Control and Prevention (CDC) from MTBC isolates of patients in the United States from 2004 to 2008 [20]. There are 2841 unique spoligotypes and 4648 unique MIRU patterns. The strains are labeled by major lineages: East Asian (Beijing), East-African Indian (CAS), Euro-American, Indo-Oceanic, *M. africanum* and *M. bovis*.

4.3.2 Most parsimonious forest generation

We used both spoligotypes and MIRU patterns to simulate the evolution of DR loci reflected in spoligotype changes. We assumed that convergent evolution is rare, and loss of spacers is irreversible. We used three distance measures for strain comparison to generate the most parsimonious forest, which is the spoligoforest of MTBC strains.

4.3.2.1 Assumptions

Mutations in the DR region involve deletion of contiguous spacers. Acquisition of additional spacers is not observed [89, 104, 128]. We call this the contiguous deletion assumption. We also hypothesize in our model that convergent evolution does not occur. This is in accordance with studies of the MTBC genome which show that homoplasy is observed rarely [104]. Using this hypothesis, we select the set of most likely parents for each spoligotype in the spoligoforest.

4.3.2.2 Distance measures for strain comparison

We used three distance measures based on two biomarkers of MTBC. Let $\vec{s}$ be 43-bit binary vector representing the spoligotype of an MTBC strain, and $\vec{m}$ be 12-bit vector representing 12-loci MIRU pattern of an MTBC strain. The biomarkers of MTBC are in the following format:

- $\vec{s}_i \in \{0, 1\}$, where $i \in \{1, .., 43\}$
- $\vec{m}_j \in \{0, .., 15\} \cup \{s, t, .., z\}$, where $j \in \{1, .., 12\}$ [1] .

We defined three distance measures based on spoligotypes and MIRU patterns: Hamming distance between spoligotypes, Hamming distance between MIRU patterns, and L1 distance between MIRU patterns. Given two spoligotypes $\vec{s}_i$ and $\vec{s}_j$, the Hamming distance between them is defined as the number of spacers that differ:

$$H_S(\vec{s}_i, \vec{s}_j) = \sum_{r=1}^{43} \mid \vec{s}_{ir} - \vec{s}_{jr} \mid$$

where $\vec{s}_{ir}$ represents the presence of spacer r of spoligotype $\vec{s}_i$. Similarly, the Hamming distance between MIRU patterns is defined as the number of MIRU loci with different number of tandem repeats:

$$H_M(\vec{m}_i, \vec{m}_j) = \sum_{r=1}^{12} \mid sign(\vec{m}_{ir} - \vec{m}_{jr}) \mid$$

where $\vec{m}_{jr}$ represents the number of repeats at MIRU locus r of 12-loci MIRU pattern $\vec{m}_i$. To highlight the difference in the number of tandem repeats at each MIRU locus, we also defined the L1 distance between MIRU patterns:

$$L_M(\vec{m}_i, \vec{m}_j) = \sum_{r=1}^{12} \mid \vec{m}_{ir} - \vec{m}_{jr} \mid \ .$$

In the spoligoforest, each spoligotype is associated with one or more MIRU patterns. Therefore, we calculate the Hamming distance and L1 distance between the MIRU

[1] The letters $\{s, t, .., z\}$ correspond to repeats with an additional mutation of 7 to 0 repeats respectively. Therefore, to separate these repeat values from the ones with numeric representation, the number of repeats $\{s, t, .., z\}$ are considered equivalent to 107 to 100 repeats respectively.

patterns of spoligotypes as the minimum of distance values between sets of MIRU patterns associated with the two spoligotypes as follows:

$$H_M(\vec{s}_i, \vec{s}_j) = \min_{\substack{\vec{m}_k \in s_i \\ \vec{m}_l \in s_j}} H_M(\vec{m}_k, \vec{m}_l)$$

$$L_M(\vec{s}_i, \vec{s}_j) = \min_{\substack{\vec{m}_k \in s_i \\ \vec{m}_l \in s_j}} L_M(\vec{m}_k, \vec{m}_l)\ .$$

4.3.2.3 Validation of the model with segregation accuracy

Based on the assumption of negligibly infrequent convergent evolution, the task of generating a mutation model of spoligotypes reduces down to finding a unique parent spoligotype for each spoligotype, if a parent exists. First, we use the contiguous deletion assumption to find a set of candidate parent spoligotypes which may be immediate ancestors of the child spoligotype. Second, we use the three distance measures defined above to find the most parsimonious forest. There are six possible permutations of these distance measures. We used segregation accuracy to find the one which leads to most parsimonious spoligoforest. Segregation accuracy is defined as the percentage of within-lineage mutation events:

$$S = \frac{\sum\limits_{l_i = l_j} d_{ij}}{\sum d_{ij}}$$

where d_{ij} is an indicator of a deletion event in which parent spoligotype $\vec{s}_i$ mutates into child spoligotype $\vec{s}_j$, and l_i represents the major lineage of MTBC strains with spoligotype $\vec{s}_i$. Maximum segregation accuracy is attained when the distance measures are used in the following order to pick the only parent among possible candidate parent spoligotypes: Hamming distance between MIRU patterns (H_M), Hamming distance between spoligotypes (H_S), L1 distance between MIRU patterns (L_M). Finally, if multiple parents still exist, a single parent is chosen at random from the set of parent candidates. The flowchart of steps to pick a single parent for each spoligotype is shown in Figure 4.2.

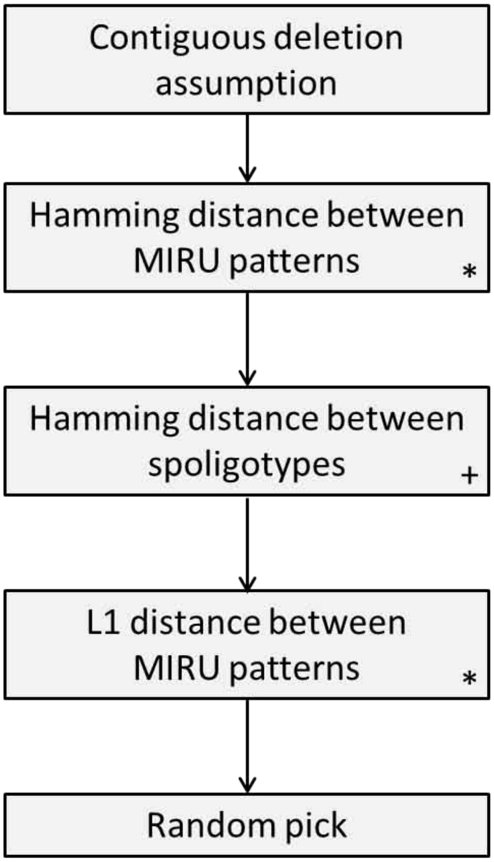

Contiguous deletion assumption
Hamming distance between MIRU patterns
*
Hamming distance between spoligotypes
+
L1 distance between MIRU patterns
*
Random pick

Figure 4.2: Flowchart of steps to pick the single parent for each spoligotype in the `MakeSpoligoforest()` algorithm using both spoligotypes and MIRU patterns. First, candidate parent spoligotypes are found based on the contiguous deletion assumption to ensure spacers in the DR region are only lost, but not gained. Then, to disambiguate the candidate parent spoligotypes, the Hamming distance between MIRU patterns, the Hamming distance between spoligotypes, and the L1 distance between MIRU patterns are used, resulting in minimum evolutionary change. Finally, if there are multiple parents still, a single parent spoligotype is picked from among the candidates at random. The spoligotype only variation of the algorithm skips the steps denoted with *, and the MIRU pattern only variation skips the steps denoted with +.

4.3.2.4 The algorithm

Based on the flowchart in Figure 4.2, we generate the most parsimonious spoligoforest using Algorithm 1. Among all candidate parent spoligotypes, we first pick the parent spoligotypes that conform to the contiguous deletion assumption. Then, we reduce the size of the candidate parent set based on maximum parsimony using three distance measures in the following order: Hamming distance between MIRU patterns, Hamming distance between spoligotypes, L1 distance between MIRU patterns. Finally, if there are multiple parents still, we pick the parent spoligotype at random. We used variations of this algorithm. If only spoligotyping is used, steps 4 and 6 are skipped in Algorithm 7. If only MIRU typing is used, step 5 is skipped in the algorithm. The software for the spoligoforest generator is available at http://sourceforge.net/projects/spolgenerator/. It is also available for use within the TB-Lineage tool at http://tbinsight.cs.rpi.edu/run_tb_lineage.html.

Algorithm 7 `MakeSpoligoforest(StrainDataset)`

Input: `StrainDataset` with spoligotypes and MIRU patterns.

Output: Spoligoforest $G = (V, E)$, where node set V represents spoligotypes, and edge set E represents spoligotype mutations.

```
1:  $E(G) = \emptyset$
2:  for each node $s \in V(G)$ do
3:      Find the set of candidate parents $P$ for node $s$ using the contiguous deletion assumption.
4:      Find $P' \subseteq P$ with the minimum Hamming distance between MIRU patterns. Set $P = P'$.
5:      Find $P' \subseteq P$ with the minimum Hamming distance between spoligotypes. Set $P = P'$.
6:      Find $P' \subseteq P$ with the minimum L1 distance between MIRU patterns. Set $P = P'$.
7:      if $|P| > 1$ then
8:          Pick a node $p \in P$ at random.
9:      else
10:         Pick the only node $p \in P$.
11:     end if
12:     Assign node $p$ as the unique parent of node $s$.
13:     Add the edge $e_{ps}$ from node $p$ to node $s$.
                $E = E \cup \{e_{ps}\}$.
14: end for
```

4.3.3 Statistical analysis of power law distributions

We observed power law distributions in the topology of spoligoforests, and tested the goodness-of-fit of these distributions. Power law distributions are often observed in the topological and graph-theoretical attributes of biological networks [135]. However, there is no single method widely accepted by scientific community for fitting power law distributions [136]. We adopt the method of analyzing power law distributions proposed by Clauset et al. [137]. According to this method, a power law distribution function is of the form:

$$p(x) = c\,x^{-\alpha}; \quad x \geq x_{min}$$

where α is the power law exponent, c is the normalization constant, and x_{min} is the lower bound for which the power law distribution holds. We also modified this function to fit a discrete power law distribution within a finite range. The method of maximum likelihood is used to estimate the exponent α. To find the lower bound x_{min}, the Kolmogorov-Smirnov statistic is used to measure the maximum distance between the cumulative distributions of the data and fitted power law model. The x_{min} value which minimizes this distance is selected as the lower bound. Finally, to test the goodness-of-fit of the power law distribution, we generate synthetic datasets from a true power law distribution using the same parameters, and compute the Kolmogorov-Smirnov statistic for each synthetic dataset relative to best-fit power law for that dataset. We calculate the p-value as the fraction of synthetic datasets for which Kolmogorov-Smirnov statistic is larger than the one observed for empirical data. If $p \geq 0.1$, then the power law distribution is a plausible fit to the data.

4.4 Results

We generated the most parsimonious spoligoforest of the CDC dataset using Algorithm 1. The spoligoforest of the CDC dataset is shown in Figure 4.1. The spoligoforest shows the spoligotypes of MTBC strains, and a putative history of mutation events in the DR region reflected in spoligotype changes. Each node in the spoligoforest represents a set of MTBC strains with a distinct spoligotype. Each edge represents a potential mutation from parent spoligotype to child spoligotype.

Table 4.1: Comparative analysis of mutation models of spoligotypes. All four models lead to high segregation accuracy, while `MakeSpoligoforest()` algorithm using both spoligotyping and MIRU-VNTR typing results in slightly higher segregation accuracy and maximizes the number of within-lineage mutation events, but the differences are not statistically significant.

Model	Segregation accuracy	# Isolated nodes	# Mutation events
Zipf model [89]	0.9921	235	2562
`MakeSpoligoforest()` (Spoligotyping)	0.9906	230	2562
`MakeSpoligoforest()` (MIRU typing)	**0.9941**	233	2562
`MakeSpoligoforest()` (Spol+MIRU)	**0.9941**	232	2562

There are 2841 nodes and 2562 edges in the spoligoforest. 2547 of these edges represent a mutation event within the same major lineage, and the segregation accuracy is 0.9941. Among 2841 nodes, 232 of them are orphan nodes, i.e. nodes with no parent or child nodes.

We compare the mutation model to existing mutation models of spoligotypes and verify that our mutation model leads to as many within-lineage mutation events as that of other mutation models. We observed interesting patterns on the topological properties of CDC spoligoforest and variations in DR loci. First, the number of descendant spoligotypes follows a power law distribution. Second, the mutation length frequency does not follow a power law. This contradicts the result of Reyes et al. that the mutation length frequency of spoligotypes follows a Zipf model [89]. Third, the number of deletion events at each contiguous DR loci follows a spatially bimodal distribution. According to this distribution, spacers 13 and 40 are identified as hotspots, and spacer 34 is the change point which is hypothesized to be rarely exposed to mutation due to deletion of the spacer earlier in the DR evolution, rather than low mutation rate. Based on this spatially bimodal distribution, we built the Starting Point Model and Longest Block Model for mutation length frequency distribution. Details for each of these results are provided in the following subsections.

4.4.1 Comparison to existing mutation models

Various models are used to generate a putative mutation history of spoligotypes. The Zipf model by Reyes et al. is based on their observation that length

Table 4.2: Candidate power law distributions and their goodness-of-fit test results based on Kolmogorov-Smirnov test. Number of descendant spoligotype frequency follows a power law distribution. On the other hand, the two power law distribution fits in the range $[8, \infty]$ and $[1, 43]$ are not plausible fits to mutation length frequency, as suggested by low p-values.

Attribute	Power law distribution function	Domain	p-value	Support for power law
D: Number of desc. spol.	$P(D = d) = 1.6906\, d^{-2.0565}$	$d \geq 2$	0.6330	Good
L: Mutation length	$P_1(L = l) = 152.9498\, l^{-3.1020}$	$l \geq 8$	0.0020	None
	$P_2(L = l) = 0.5108\, l^{-1.6963}$	$1 \leq l \leq 43$	0	None

of unambiguous mutation events follows a Zipf distribution, and they generate spoligoforests based on this probabilistic model [89, 138]. We used an independent biomarker, MIRU, to disambiguate the possible ancestors for each spoligotype. We compared the Zipf model to our model, `MakeSpoligoforest()` algorithm, with three variations: using spoligotyping only, using MIRU-VNTR typing only, and using spoligotyping and MIRU-VNTR typing as shown in Figure 4.2 in combination with the original version of the algorithm described in the methods section. Table 4.1 shows the comparative analysis of the resulting spoligoforests based on these four models. In the spoligoforests of all four models, there are 2562 mutation events, represented by edges in the spoligoforest. Isolated nodes in the spoligoforests are the nodes with no parent or child spoligotype, so their ancestor and descendant spoligotypes are unidentified from the mutation history. The number of isolated nodes differs slightly. Segregation accuracy is the highest in the spoligoforest based on `MakeSpoligoforest()` algorithm using both spoligotyping and MIRU-VNTR typing, and equal to the segregation accuracy of the spoligoforest generated using the variation of `MakeSpoligoforest()` algorithm with MIRU typing. However, the difference between the segregation accuracy of different mutation models is not statistically significant. This validates that simpler spoligoforest models based only on the length of deletion can be used with little or no degredation in the results.

4.4.2 Number of descendant spoligotypes

Each spoligotype can have at most one parent spoligotype, and any number of child spoligotypes, assuming convergent evolution does not occur. A single mutation event in the DR region results in a new child spoligotype. The number of immediate

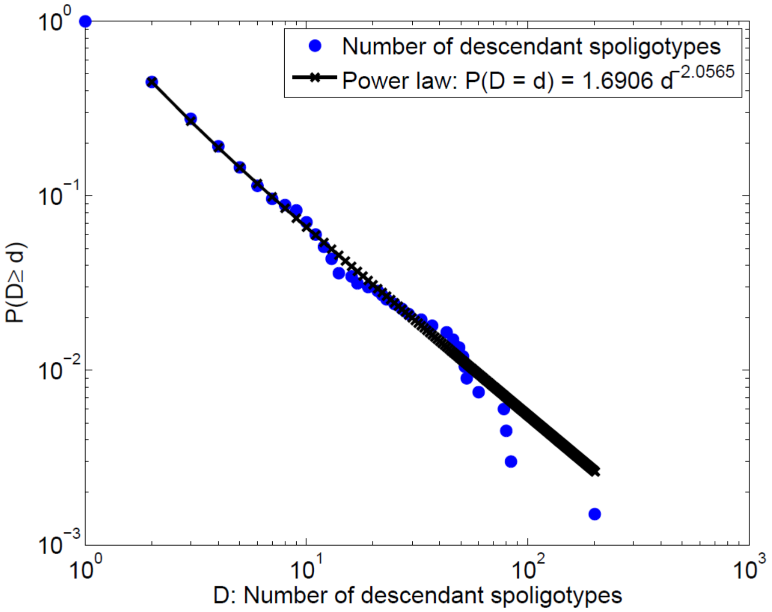


Figure 4.3: Number of descendant spoligotypes follows a power law distribution, which holds for spoligotypes with $d \geq 2$ children spoligotypes.

descendant spoligotypes for each spoligotype depends on the number of spacers present in the DR region, which is equivalent to the copy number of the spoligotype. In theory, the copy number of a spoligotype can range from 0 to 43, but not all spoligotype representations were observed in the dataset we analyzed. Let d_i be the number of descendants of the spoligotype represented by node s_i. Figure 4.3 shows the cumulative distribution of descendant spoligotype count frequency $P(D \geq d)$ on a log-log plot. We used the power-law fitting procedure introduced by Clauset et al. [137] to test whether the data follows a power law distribution. Table 4.2 shows the power law distribution function fit to the number of descendant spoligotypes. The power law distribution holds for all spoligotypes with $d \geq 2$ children spoligotypes. Based on the Kolmogorov-Smirnov test, the p-value of 0.6330 is larger than 0.1, which suggests that a power law is a plausible fit to the number of descendant spoligotypes. The power law observation is based on two facts: 1) the higher the copy number of the spoligotype, the more descendants it can have, 2) the number of descendant spoligotypes increases due to the assumption of no convergent evolution, which leads to higher genetic diversity. The number of descendants of a spoligotype can also be interpreted as the number of one-step deletion events that lead to new spoligotypes.

4.4.3 Mutation length

Mutation length is defined as the number of contiguous spacers deleted in a mutation event. According to the contiguous deletion assumption, at each mutation event, a set of contiguous spacers can only be deleted, but not gained [89, 104]. In theory, mutation length can range from 1 to 43. Reyes et al. used only unambiguous deletion events in cluster-graphs, and observed that mutation length frequency follows a Zipf distribution [89]. Based on the putative mutation history of spoligoforest for the CDC dataset, we checked if a power law distribution is a plausible fit to mutation length frequency. We used the same procedure introduced by Clauset et al. to test whether the mutation length follows a power law distribution [137]. Let l_{ij} be the length of mutation event from node s_i to node s_j. Figure 4.4 shows the cumulative distributions $P(L \geq l)$ of two candidate power law distribution fits to the mutation length on a log-log plot. Table 4.2 shows two power law distribution function fits to mutation length, one in the range $[1, \infty]$, and one in the range $[1, 43]$. The first power law distribution holds only for the mutation events with length $l \geq 8$. Among all 2562 mutation events represented by edges in the spoligoforest, only 263 of them, which constitute 10.27% of all mutation events, are of length $l \geq 8$. Therefore, power law distribution does not fit most of the observed mutation events. Moreover, based on the Kolmogorov-Smirnov test, the p-value of 0.0020 is smaller than 0.1, which suggests that this power law distribution is not a plausible fit to mutation length.

The second power law distribution fit in the range $[1, 43]$ has the probability mass function of the form used in Reyes et al. to fit the Zipf distribution [89]:

$$P_2(L = l) = \frac{l^{-\alpha}}{\displaystyle\sum_{i=1}^{43} i^{-\alpha}} .$$

The resulting p-value of the distribution based on the Kolmogorov-Smirnov test is 0, and the second power law distribution is also not a plausible fit to mutation length.

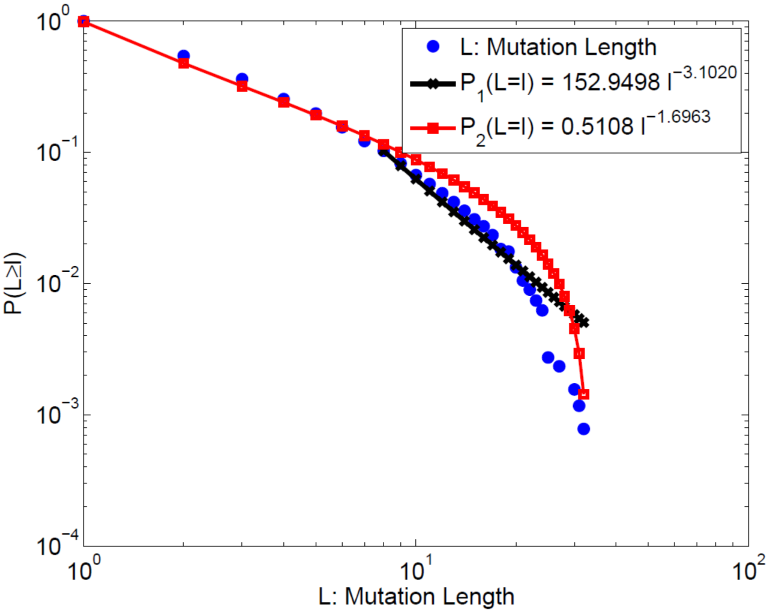


Figure 4.4: Mutation length frequency does not follow a power law distribution. The Kolmogorov-Smirnov test indicates that both power law distributions do not hold.

Therefore, mutation length does not follow a power law distribution, in contrast to the results in Zipf model built by Reyes et al. On the other hand, it is still accurate to claim that observed mutation patterns involve high numbers of short spacer deletions and small numbers of long spacer deletions. This is because mutation length depends on the number of contiguous spacers in the parent spoligotypes.

4.4.4 Number of mutations at each DR locus

We counted the number of mutations which result in deletion of each spacer to identify variations of the mutation rates in the DR region. Figure 4.5 shows the number of mutation events in which a spacer of each DR locus is deleted. Based on this figure, the number of deletions for each spacer follows a spatially bimodal distribution, and the modes are spacer 13 and spacer 40. We call these DVR regions *hotspots*, or sites of increased observed variability. Spacer 34 is the change point in the bimodal distribution. This is due to lack of spacer 34 in the DR region in most MTBC strains. In fact, out of 2841 spoligotypes in the CDC dataset, only 94 of them, which constitute 3.31% of all spoligotypes, have spacer 34 present in the DR region. Out of 9336 MTBC strains determined by spoligotype and 12-loci MIRU patterns in the dataset, only 192 of them, which constitute 2.06% of all MTBC strains, have spacer 34 present in the DR region. Therefore, the mutation rate is lowest at spacer 34 due to absence of spacer 34 in most MTBC strains in the CDC dataset.

Figure 4.6 shows the CDC spoligoforest colored by presence of spacer 34. The spoligoforest is dominated by nodes in gray, which denote the spoligotypes with spacer 34 absent, and the nodes in blue, representing the spoligotypes with spacer 34 present, are very few. This suggests that spacer 34 have been irreversibly deleted in the early stages of DR evolution and they can not mutate further after being deleted. Two out of three principal genetic groups defined by Sreevatsan et al., PGG2 and PGG3, lack spacers 33 to 36, which is concordant with this observation [139]. In addition, 1971 spoligotypes out of 2841 in the CDC dataset are labeled with Euro-American lineage, which is characterized by the deletion of spacers 33-36 [4,45]. This bimodal separation of DR loci leads to accumulation of shorter deletions among mutation events, rather than observing longer deletions, which explains why the power law distribution is actually not a plausible fit to mutation length.

4.4.5 Alternative models for mutation length frequency

We showed earlier that a power law distribution is not a plausible fit to mutation length frequency. Looking at the number of mutations at each spacer shown in Figure 4.5, we can see that the mutation length depends on the starting point of the mutation. Based on this observation, we built two alternative models for mutation length frequency: Starting Point Model (SPM) and Longest Block Model (LBM). SPM conditions the mutation length on the starting point of the mutation, and fits power law distributions to mutation length frequency. LBM conditions the mutation length on the length of longest block of 1's, or present spacers, beyond the starting point of mutation on each mutated spoligotype. In this section, we describe both models in detail and show that they are plausibly good fits to mutation length frequency.

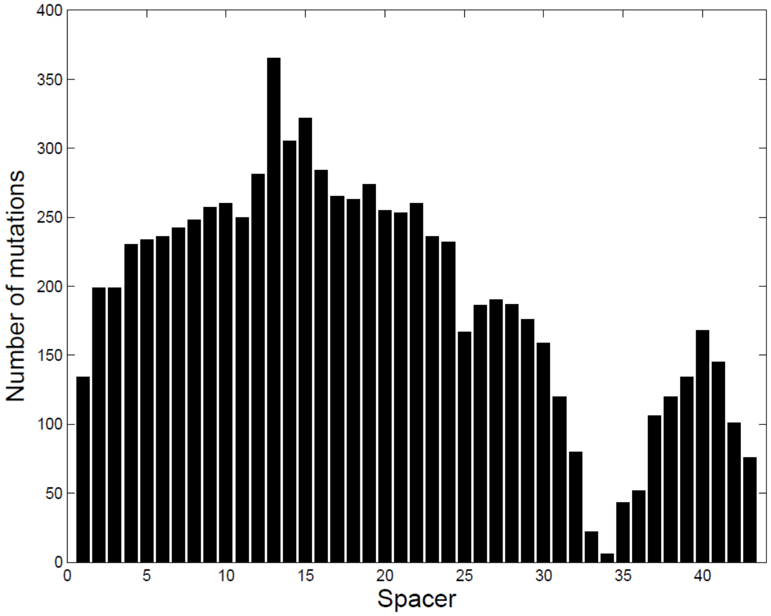


Figure 4.5: Number of deletions at contiguous DR loci follows a spatially bimodal distribution. The two modes are spacer 13 and spacer 40, which are hotspots. The change point is spacer 34.

Table 4.3: Possible start and end point regions of mutation events. The table shows that a mutation event can start and end either in region 1 including spacers [1, 34], leading to Type I mutation event, or it can start and end in region 2 including spacers [35, 43], leading to Type II mutation event.

Start	End	Possible?
[1, 34]	[1, 34]	✓
[1, 34]	[35, 43]	✗
[35, 43]	[1, 34]	✗
[35, 43]	[35, 43]	✓

4.4.5.1 Starting Point Model (SPM)

The number of mutations follows a spatially bimodal distribution as shown in Figure 4.5, and spacer 34 is the change point of this distribution with the fewest number of mutations. We relax this assumption and separate the spacers into two regions: region 1 including spacers [1,34], and region 2 including spacers [35,43]. Without loss of generality, we assume that a mutation event starts at a lower-indexed spacer and ends at a higher-indexed spacer. Table 4.3 shows all start and

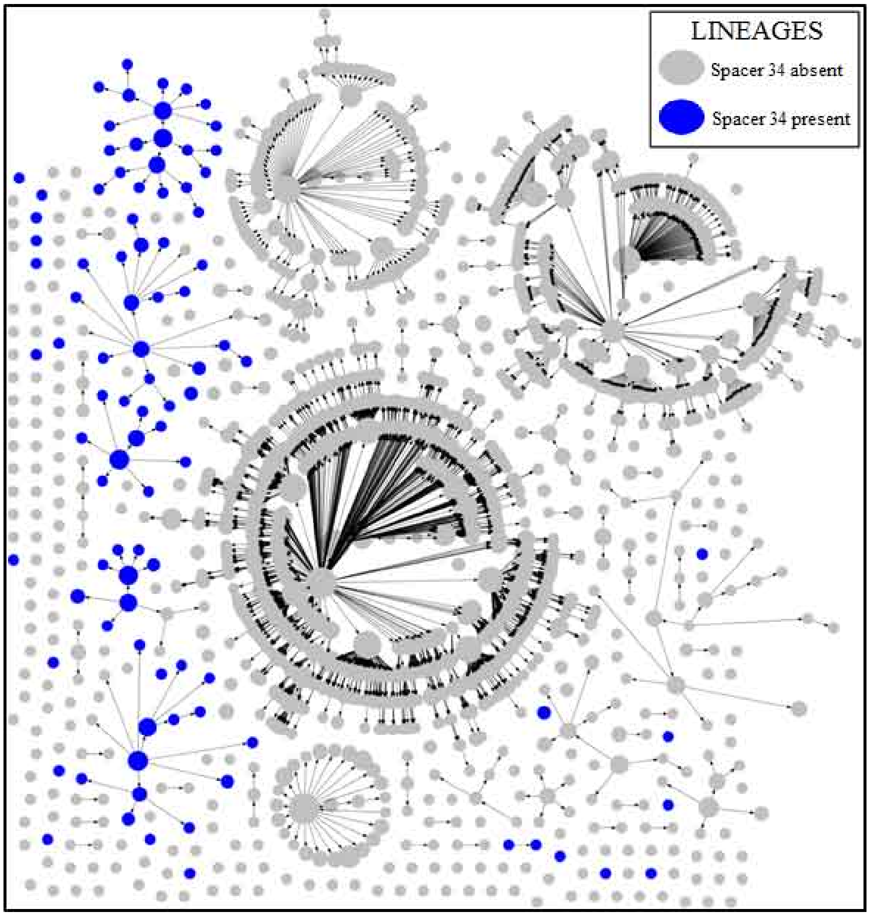


Figure 4.6: The spoligoforest of the CDC dataset colored by the presence of spacer 34. Gray nodes which represent spoligotypes with spacer 34 absent dominate the spoligoforest, compared to the blue nodes which represent spoligotypes with spacer 34 present. This is because spacer 34 has been deleted irreversibly in most of the spoligotypes earlier in the DR evolution, thus can not mutate further.

end point combinations of mutation events, and whether they are possible or not. According to the table, both start and end points of a mutation event have to be in the same spacer region, either in [1,34], or in [35,43]. Therefore, no mutation event can result in deletion of spacers in both regions. We name mutation events which start and end at region 1 in the range [1,34] as *Type I mutation event*, and mutation events which start and end at region 2 in the range [35,43] as *Type II mutation event*.

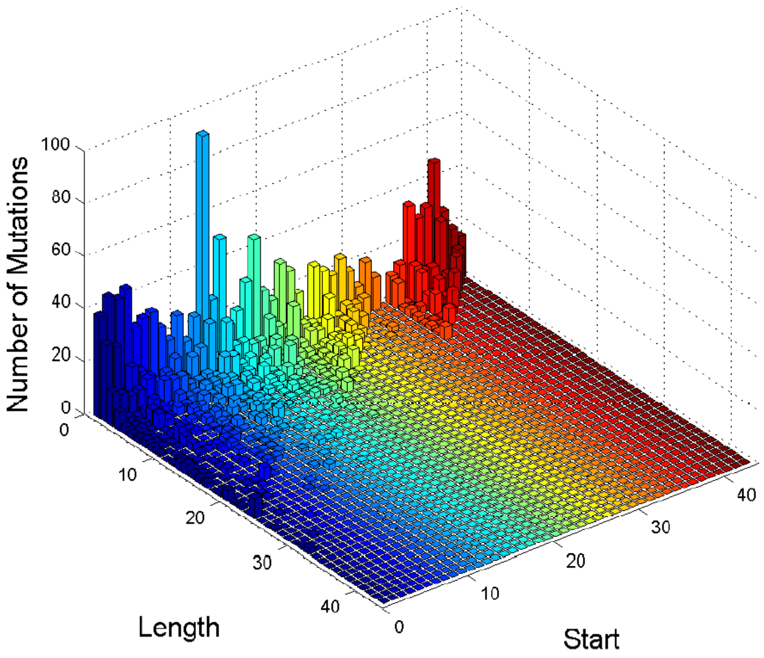


Figure 4.7: The number of mutations of length $L = l$ which start at spacer $S = s$. For each starting point except 34 and 43, mutation length frequency follows a power law distribution, as verified by a goodness-of-fit test. For starting points 34 and 43, the only possible mutation length is 1.

$$\boxed{P(L = l|S = s) = \begin{cases} \dfrac{l^{-\alpha_s}}{\displaystyle\sum_{i=1}^{35-s} i^{-\alpha_s}}, & s \in [1, 33] \\ \dfrac{l^{-\alpha_s}}{\displaystyle\sum_{i=1}^{44-s} i^{-\alpha_s}}, & s \in [35, 42] \\ 1, & s \in \{34, 43\}, l = 1. \end{cases}} \tag{4.1}$$

Let variable L represent the mutation length, and variable S represent the starting point of mutation. Figure 4.7 shows the number of mutations of length $L = l$ which start at spacer $S = s$. Notice that, at each starting point except spacers 34 and 43, the mutation length frequency follows a power law distribution, as verified by goodness-of-fit test. Using maximum likelihood estimation, we verified that the mutation length frequency follows a power law distribution with a unique

Table 4.4: Candidate models for mutation length frequency distribution and their goodness-of-fit test results based on Kolmogorov-Smirnov test. The *p*-value for both SPM and LBM is 1, which suggests that both models estimate the mutation length frequency distribution accurately.

Model	KS-value	p-value	Support
SPM	0.0406	1	Good
LBM	0.0116	1	Good

power law exponent for each starting point $S = s \in [1, 33] \cup [35, 42]$. At the boundary starting spacers 34 and 43, since the mutation event can occur in one of the two regions, the only possible mutation length is 1. These observations lead to the SPM described in Equation (4.1), where α_s is the exponent of power law distribution for starting point s. The α_s values are estimated from the CDC data using maximum likelihood method to compute the $P(L = l|S = s)$ values. In order to find the mutation length distribution, $P(L = l)$ values are calculated as follows:

$$P(L = l) = \sum_{s=1}^{43} P(L = l|S = s)\ P(S = s) \tag{4.2}$$

where $P(S = s)$ is calculated as follows. For each spoligotype in the dataset, present spacers are found and added to the total count for each spacer. Then, $P(S = s)$ is the ratio of the number of spoligotypes with spacer s present to the total number of spacers present in all spoligotypes. Note that $P(S = s)$ values are derived from the spoligotype signatures of strains, without the need to find the mutation history of spoligotypes using `MakeSpoligoforest()` algorithm.

Given $P(L = l)$ values calculated using Equation (4.2), Figure 4.8 shows the cumulative distribution $P(L \geq l)$ for the SPM on a log-log plot. In order to test the goodness-of-fit of SPM, we adapted the power law validation procedure by Clauset et al. to this model. Table 4.4 shows that the p-value for SPM is 1, which is greater than 0.1, which suggests that SPM is a plausible model for the mutation length frequency distribution.

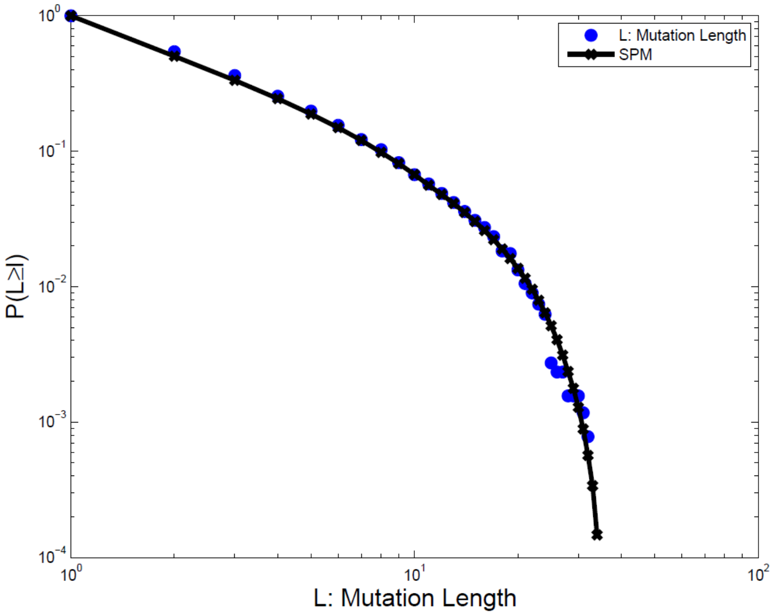


Figure 4.8: Cumulative distribution $P(L \geq l)$ of SPM based on the CDC dataset. SPM is a good model for mutation length frequency distribution. The goodness-of-fit test returns a p-value of 1, which verifies the accuracy of the model.

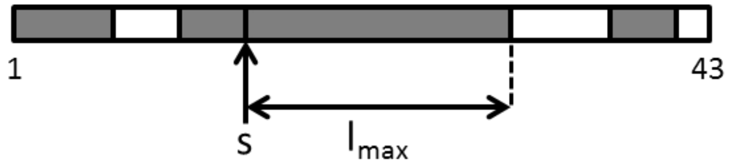


Figure 4.9: Given the starting point of mutation on a spoligotype, the contiguous block of spacers beyond the starting point can be deleted in a mutation event. In the example above, given that the mutation starts at spacer s, the length of the mutation can be at most l_{max}.

$$\boxed{P(L = l | L \leq l_{max}) = \begin{cases} \dfrac{l^{-\alpha_{l_{max}}}}{\displaystyle\sum_{i=1}^{l_{max}} i^{-\alpha_{l_{max}}}}, & l_{max} > 1, P(L \leq l_{max}) \neq 0 \\ 1, & l_{max} = l = 1. \end{cases}} \tag{4.3}$$

4.4.5.2 Longest Block Model (LBM)

According to the contiguous deletion assumption, a block of contiguous spacers can be deleted in a mutation event. Therefore, given the starting point of a mutation on a spoligotype, the number of spacers that can be deleted is limited by the number of contiguous spacers beyond the starting point. Let L represent the mutation length, and let l_{max} be the upper bound on the mutation length, given the starting point of the mutation on a spoligotype. Figure 4.9 shows a spoligotype with present spacers in gray, and absent spacers in white. If the starting point of mutation is s, then the mutation length can be at most l_{max}. Based on this observation, in Figure 4.10, we plotted the number of mutations of length $L = l$, given the maximum possible length l_{max}. In the plot, at each l_{max} value except $l_{max} = 1$, the mutation length frequency follows a power law distribution if $P(L \leq l_{max}) \neq 0$, also verified by goodness-of-fit test. We verified using maximum likelihood estimation that the mutation length frequency follows a power law distribution with a unique power law exponent for each $l_{max} > 1$, given that $P(L \leq l_{max}) \neq 0$. At the boundary case $l_{max} = 1$, the mutation can only be of length 1. These observations are combined in the LBM described in Equation (4.3), where $\alpha_{l_{max}}$ is the exponent of power law distribution for maximum length l_{max}. The $\alpha_{l_{max}}$ values are estimated from the CDC data using maximum likelihood method, and $P(L = l|L \leq l_{max})$ values are found. Mutation length distribution is derived from this probability as follows:

$$P(L = l) = \sum_{l_{max}=1}^{43} P(L = l|L \leq l_{max})\ P(L \leq l_{max}) \tag{4.4}$$

where $P(L \leq l_{max})$ values are calculated from the mutation history of spoligotypes using starting point of each mutation and thereby the length of the longest block of contiguous spacers for each mutation event.

Given $P(L = l)$ values calculated using Equation (4.4), Figure 4.11 shows the cumulative distribution $P(L \geq l)$ for LBM on a log-log plot. We tested the goodness-of-fit of LBM using the test we adapted from power law validation procedure by Clauset et al. As shown in Table 4.4, the p-value of the test for LBM is 1, which is greater than 0.1. This suggests that LBM is a plausible model for the mutation

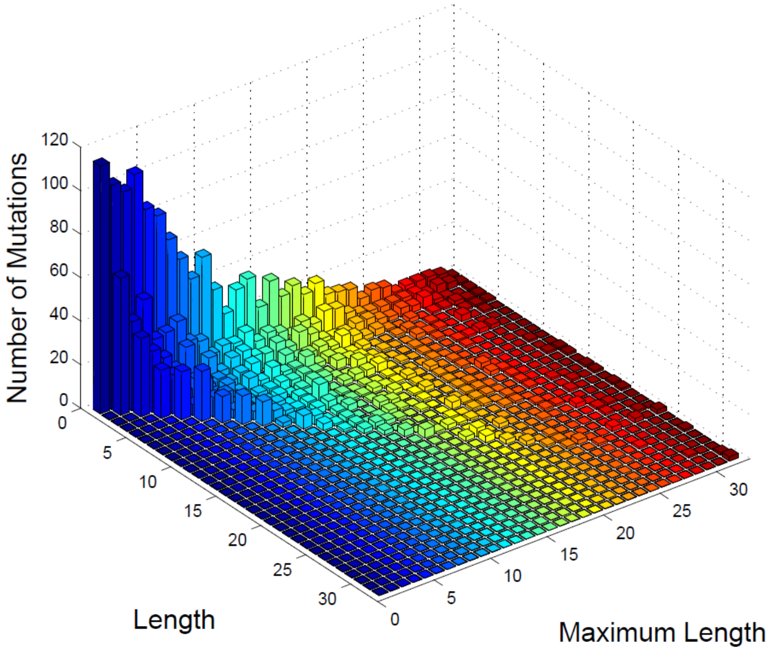


Figure 4.10: The number of mutations of length $L = l$, given that the longest block of contiguous spacers is of length $L = l_{max}$. If there exists at least one block of contiguous spacers of length $l_{max} > 1$, then, given the upper bound l_{max}, the mutation length frequency follows power law distribution.

Table 4.5: Goodness-of-fit test results of SPM and LBM for the dataset from Institut Pasteur de Guadeloupe, based on Kolmogorov-Smirnov test. The p-value for SPM and LBM is 1, which suggests that both models estimate the mutation length frequency distribution accurately. This shows that SPM and LBM are robust and they hold for different strain datasets.

Model	KS-value	p-value	Support
SPM	0.0482	1	Good
LBM	0.0381	1	Good

length frequency distribution.

In both SPM and LBM, we used an extra parameter to estimate the mutation length frequency distribution. To test if the models are robust, we applied SPM and LBM to another dataset from Institut Pasteur de Guadeloupe which is partially listed in multimarker SITVITWEB database [140]. This dataset has 2158 strains uniquely identified by (spoligotype, MIRU) pairs, and there are 699 unique spoligotypes. We ran the `MakeSpoligoforest()` algorithm on this dataset, and fit

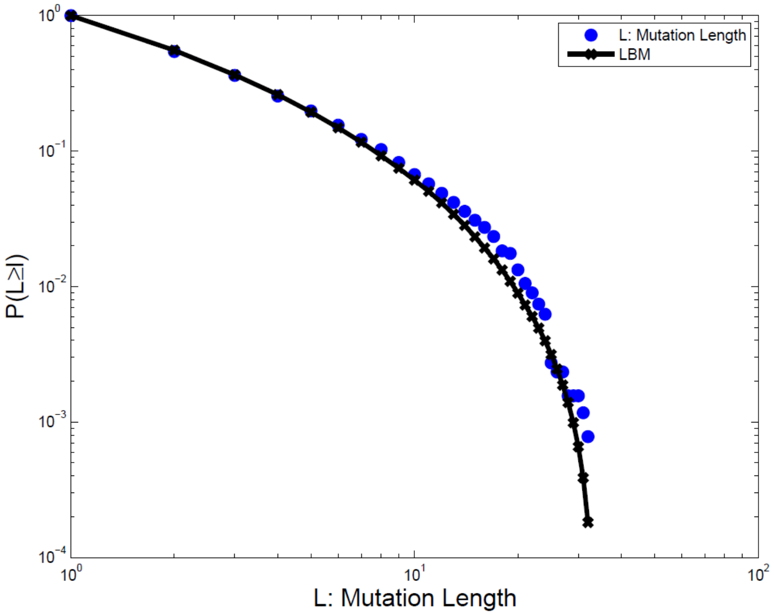
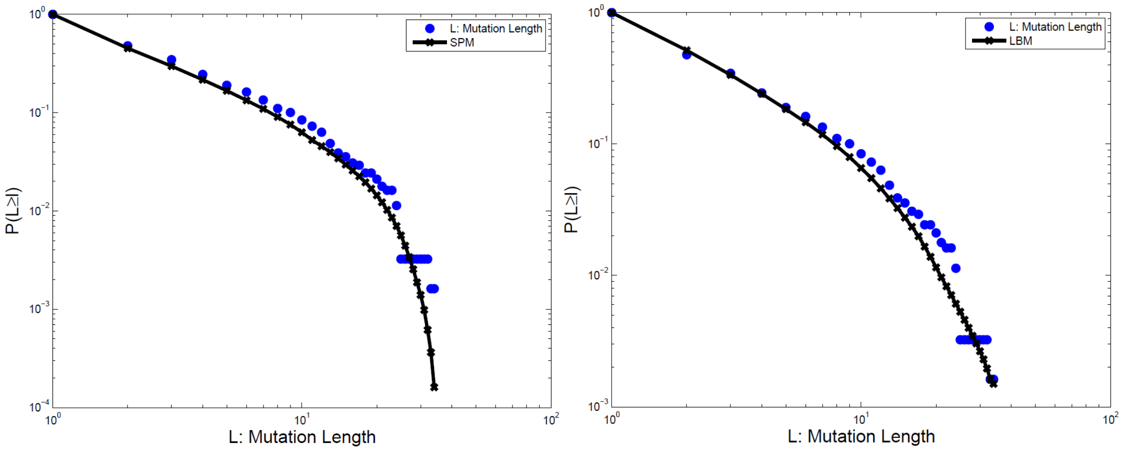


Figure 4.11: Cumulative distribution $P(L \geq l)$ of LBM based on the CDC dataset. LBM is a good model for mutation length frequency distribution. The goodness-of-fit test returns a p-value of 1, which verifies the accuracy of the model.



(a) SPM: Starting Point Model

(b) LBM: Longest Block Model

Figure 4.12: Cumulative distribution $P(L \geq l)$ for SPM and LBM based on the Institut Pasteur de Guadeloupe dataset. SPM and LBM fits the mutation length frequency distribution. This shows that these models are robust and they hold for different strain datasets.

SPM and LBM to the mutation length frequency distribution. Figure 4.12a and 4.12b shows the cumulative distribution $P(L \geq l)$ for SPM and LBM respectively. The goodness-of-fit test results for both models are summarized in Table 4.5. The p-value for both models is 1, which is greater than 0.1, and this suggests that SPM and LBM are plausibly good models for the mutation length frequency distribution. Therefore, SPM and LBM are robust and they accurately estimate the mutation length frequency distribution independent of the dataset examined.

4.5 Discussion and Conclusion

We developed a new mutation model of MTBC spoligotype evolution using the variations in the DR region and MIRU patterns to disambiguate the ancestors of a spoligotype. Based on the contiguous deletion assumption and no homoplasy, and using three distance measures, we generated the most parsimonious forest of spoligotypes. The resulting spoligoforest depicts a putative history of mutation events in the DR region. Given the spoligotype mutations, we analyzed the biological network of spoligotypes in terms of both network topology and number of mutations at each DR locus.

We compared our mutation model based on spoligotypes and MIRU patterns with its counterparts using spoligotyping only, MIRU typing only and with Zipf model [89]. The mutation model which incorporates both biomarkers results in the most parsimonious spoligoforest and maximizes within-lineage mutation events. The comparison showed that segregation accuracy values are high in all four models with no statistically significant difference in the results. Therefore, spoligoforests created using only spoligotypes and the Zipf model are very similar to spoligoforests determined by the additional independent biomarker MIRU-VNTR. This validates the spoligoforest algorithms based only on spoligotypes, showing that spoligotype only algorithms can be used to generate the spoligoforest when MIRU patterns are not present.

The number of descendants of a spoligotype is equivalent to the outdegree of the corresponding node in the spoligoforest. We tested and verified the hypothesis that the number of descendant spoligotypes follows a power law distribution. This is due to the fact that the higher the copy number of spoligotype, that is, the more spacers present in the DR region, the more spoligotypes can descend from it. In addition, the assumption of no homoplasy favors genetic diversity rather than convergent evolution.

We tested and verified that mutation length of spoligotype deletions in a mutation event does not follow a power law distribution, as opposed to the Zipf model for mutation of spoligotypes proposed by Reyes et al. [89]. However, it is still accurate to state that mutations in the DR region rarely involve long deletions and frequently involve short deletions.

We calculated the number of mutation events which resulted in deletion of spacer at each DR locus. The number of mutations at consecutive DR loci showed a pattern of spatially bimodal distribution. The two modes are spacer 13 and spacer 40, which are hotspots of variations in the DR region. The change point in the bimodal distribution is spacer 34. This is due to absence of spacer 34 in a large number of MTBC strains, rather than low mutation rate at DVR34, because this spacer has been deleted irreversibly at the beginning of DR evolution and it can not mutate further after being deleted. Two out of three principal genetic groups defined by Sreevatsan et al. and MTBC strains of Euro-American lineage lack spacers 33-36, which supports the claim that low number of mutations in DVR34 is due to lack of spacer 34 in most MTBC strains in the CDC dataset [4, 139]. Since most of the deletion events occur either on spacers 1-34, or spacers 35-43, resulting in accumulation of shorter deletions, longer deletions are not observed. Therefore, this bimodal distribution explains why mutation length does not follow a power law distribution. Note however that a block of contiguous spacers in the 43-spacer format may not be contiguous in the 104-spacer format, which is a superset of 43-spacer format [141]. Therefore, 43-spacer representation can be renumbered on the 104-spacer format for further differentiation of spoligotypes to build a more detailed mutation history of the DR region. Spacer duplications can also intervene a block of contiguous spacers during the microevolution of genetically related group of strains, and a deletion involving a duplicated spacer can not be captured by the 43-spacer format [142].

Based on the spatially bimodal distribution of mutation events in the DR region, we built two alternative models for mutation length frequency. The SPM conditions the mutation length on the starting point of the mutation, and the LBM conditions the mutation length on the length of the longest block of contiguous spacers beyond the starting point of mutation. Both SPM and LBM estimate the mutation length frequency distribution accurately, as opposed to Zipf model suggested in earlier studies. We also tested these models on another dataset from Institut Pasteur de Guadeloupe, and verified that both models are robust and hold for different strain datasets.

Future work will involve analysis of other topological attributes of the spoligoforest, extension of the mutation model to use other biomarkers, and interpretation of clades grouped closely in the spoligoforest. The mutation model can be extended to include more biomarkers, e.g. RFLP, with corresponding distance measures for the additional biomarkers to be used in the algorithm which generates the spoligoforest. Analysis of connected components in the spoligoforest can give more insight into segregation of major lineages or sublineages. In addition, each tree or subtree in the spoligoforest can be a group of genetically related MTBC strains not classified as a separate clade earlier. This mutation model can also be extended to other organisms genotyped by CRISPR profiles.

CHAPTER 5
HOST-PATHOGEN ASSOCIATION ANALYSIS OF TUBERCULOSIS PATIENTS VIA UNIFIED BICLUSTERING FRAMEWORK

5.1 Introduction

Tuberculosis (TB) is an airborne disease which is a leading cause of death worldwide. According to World Health Organization, one third of the human population is infected either latently or actively with TB [1]. *Mycobacterium tuberculosis* complex (MTBC) is the set of species which causes TB. MTBC isolates from TB patients are genotyped using multiple biomarkers for tracking TB transmission, TB control, and examining host-pathogen relationships.

Earlier studies have found associations between TB patients and the MTBC strains which infected them. Hirsh et al. showed that a TB patient's place of birth can be used to predict the geographic origin of the MTBC isolate [31]. Gagneux et al. defined the population structure of MTBC strains using six phylogeographic lineages and showed that these lineages are adapted to particular human populations defined by place of birth or risk factor [4]. Visual inspection via host-pathogen maps enable making inferences from patient data and strain lineages [143]. Although names of phylogeographic lineages imply an association between MTBC isolates and patients' place of birth, none of these studies combine genetic proximity between MTBC strains and spatial proximity between TB patients together. In this study, in addition to the distribution of MTBC isolates to their host's country of birth, we add genetic proximity, spatial proximity and time into domain knowledge of host-pathogen association analysis.

Multiple sources of information can be incorporated into data analysis via data fusion [144]. Recently, there has been considerable work on genomic data

* Portions of this chapter previously appeared as: C. Ozcaglar, B. Yener, and K. P. Bennett, Host-pathogen association analysis of tuberculosis patients via Unified Biclustering Framework, Tech. Rep. 12-05, Department of Computer Science, Rensselaer Polytechnic Institute, 2012.

fusion [145–147]. In the TB context, Ozcaglar et al. built the tensor clustering framework (TCF) to cluster MTBC strains using multiple biomarkers simultaneously through genomic data fusion [23]. Genomic and phenomic data sources are also combined in earlier studies [148] via genome-phenome data fusion. However, there is no significant work on genome-phenome interactions of MTBC isolates and TB patients.

In this study, we present host-pathogen associations of tuberculosis by incorporating genetic proximity between MTBC strains, spatial proximity between TB patients, and time into domain knowledge via Unified Biclustering Framework (UBF). We simultaneously factorize multiple sources of information in various forms and obtain biclusters which represent host-pathogen pairs, while keeping pathogens genetically close in order to estimate most likely mutation events, and keeping hosts spatially close in order to estimate most likely transmission events. Based on factor matrices of hosts and pathogens, we generate the feature pattern similarity matrix of host-pathogen pairs, and find density-invariant biclusters. Finally, we select statistically significant biclusters among them and find the most stable host-pathogen associations. We also find host-pathogen associations within each major lineage. We evaluate biological relevance of statistically significant biclusters, confirm known host-pathogen associations, and propose new ones.

5.2 Background

In order to find relationships between MTBC isolates and TB patients, we uniquely identified them by their characteristics. We represented MTBC strains with a commonly used biomarker, spoligotype, and represented each patient with their country of birth. Finally, we stated the host-pathogen association analysis as a biclustering problem. Next, we give a brief background on spoligotyping, biclustering, and explain host-pathogen association analysis as a biclustering problem.

5.2.1 Spoligotyping

Spoligotyping is a DNA fingerprinting method of MTBC which exploits the polymorphism in the DR region consisting of 36 bp of direct repeats separated by 36 to 41 bp of spacers [17]. A spoligotype consists of 43 spacers, and it is represented as a 43-bit binary vector, where zeros represent absence of spacers and ones represent presence of spacers. Mutations in the DR region can result in loss of spacers, but not gain. This rule of irreversible mutation of spoligotypes is also known as contiguous deletion assumption [32, 33].

5.2.2 Biclustering

Biclustering is a class of clustering algorithms which perform simultaneous clustering of rows and columns of a matrix. The term was first coined by Cheng and Church for gene expression data analysis [49]. Following them, many biclustering algorithms motivated mostly by bioinformatics applications are developed. These biclustering algorithms include spectral biclustering algorithm by Dhillon et al. [54] and Kluger et al. [55], Statistical-Algorithmic Method for Bicluster Analysis (SAMBA) by Tanay et al. [58], Coupled Two-Way Clustering (CTWC) by Getz et al. [52], Binary Inclusion-Maximal biclustering algorithm (BiMax) by Prelic et al. [60], and densely-connected biclustering (DECOB) by Colak et al. [64]. A great survey by Madeira et al. details biclustering and existing biclustering algorithms for biological data analysis [48].

5.2.3 Host-pathogen association analysis: a biclustering problem

Biclustering was initially motivated by gene expression data analysis in order to group genes into subsets of genes which are coexpressed under certain subsets of conditions. This is equivalent to finding submatrices in a gene expression matrix such that the submatrix entries follows a cohesive pattern under investigation. In the TB context, the genes of microarray data maps to spoligotypes of MTBC strains, and the conditions of microarray data maps to country of birth of TB patients. The resulting host-pathogen matrix of tuberculosis expresses the association level of a spoligotype to a country.

In the case where the original host-pathogen matrix is extended or concatenated with other matrices via data fusion, we use feature patterns for spoligotypes and countries. We first extract feature patterns for each spoligotype of MTBC strains and for each country of birth for TB patients. The association level of a spoligotype and a country is calculated as the cosine similarity of their feature pattern vectors. This final form of host-pathogen matrix of tuberculosis expresses association level of host-pathogen pairs, and is in the correct form to be analyzed via biclustering.

In the next section, we present the methods used for host-pathogen association analysis. We first give details about the patient dataset. Then, we present the calculation of genetic proximity matrix and spatial proximity matrix used in data fusion. Finally, we present the steps of Unified Biclustering Framework (UBF).

5.3 Methods

5.3.1 The dataset

The NYC dataset consists of 4876 patients in the United States diagnosed between 2001 and 2007. The spoligotype of MTBC strains and their host's country of birth are available in the dataset, along with the date of diagnosis. There are 858 unique spoligotypes in the original dataset. MTBC strains are labeled by major lineages based on their spoligotypes using Conformal Bayesian Network (CBN) model [20], and by KBBN sublineages using the Knowledge-based Bayesian Network (KBBN) model [21]. We refer to spoligotypes using shared type numbers, or SIT numbers using SITVITWEB database [140]. If the spoligotype is not assigned to an ST number by SITVITWEB, then we assign a unique UST number, where U denotes unknown ST. We first filter this data such that there are at least 2 patients from each country, and at least two patients infected with each strain. After filtering the dataset, there remains 4301 patients, 311 spoligotypes, and 104 countries. Using this filtered dataset, we construct the host-pathogen tensor (HPT) of the form *Spoligotypes* × *Countries* × *Time.* The final HPT is denoted as $\underline{\mathbf{X}} \in \mathbb{R}^{(I=311)\times(J=104)\times(K=7)}$. The host-pathogen tensor (HPT) is shown in Figure 5.1.

5.3.2 Distance matrices

In the host-pathogen tensor, the first mode represents pathogen attributes, in this case spoligotypes. Genetic proximity of spoligotypes can be found using genetic distance measures. Hosts with genetically close spoligotypes are more likely to be

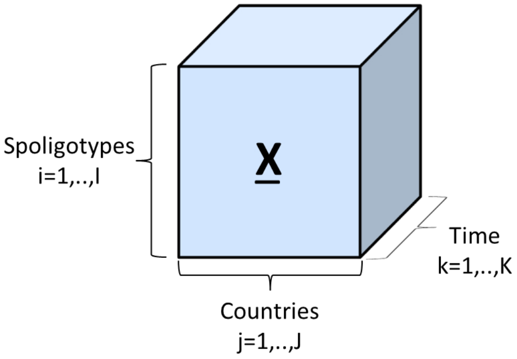


Figure 5.1: Host-pathogen tensor (HPT). The first mode represents spoligotypes, the second mode represents countries, and the third mode represents time. This HPT is of the form *Spoligotypes* × *Countries* × *Time*.

involved in the same mutation event. Similarly, the second mode represents host attributes, in this case country of birth. Proximity of countries can be found based on neighbourhood. Patients from close countries based on the proximity values are more likely to be involved in the same transmission event.

5.3.2.1 Genetic proximity matrix

Given 311 distinct spoligotypes, we define a genetic proximity measure between them. Mutation of spoligotypes is based on the Contiguous Deletion Assumption (CDA), which states that one or more contiguous spacers can be deleted in a mutation event, but not gained. Let s_i represent spoligotype i, and let $s_i \rightarrow s_j$ represent the mutation of spoligotype s_i into spoligotype s_j. Then, we define the CDA matrix, which summarizes contiguous deletion assumption, as follows:

$$\mathrm{CDA}(s_i, s_j) = \begin{cases} \text{true}, & \text{if } s_i \rightarrow s_j \text{ or } s_j \rightarrow s_i \\ \text{false}, & \text{otherwise.} \end{cases}$$

Let $H(s_i, s_j)$ be the Hamming distance between spoligotypes s_i and s_j, as defined in [32]:

$$H\left(s_i, s_j\right) = \sum_{r=1}^{43} \mid s_{ir} - s_{jr} \mid$$

where s_{ir} represents the value of $r-th$ spacer of spoligotype s_i. Then, we define the genetic proximity matrix P_G as follows:

$$P_G(s_i,s_j)=\begin{cases}\dfrac{1}{1+H(s_i,s_j)}, & \text{if } i\neq j,\ \text{CDA}(s_i,s_j),\ H(s_i,s_j)\leq 10\\ 1, & \text{if } i=j\\ 0, & \text{otherwise.}\end{cases}$$

Genetic proximity matrix P_G has values inversely proportional to the Hamming distance between two spoligotypes, as long as the Hamming distance between them is at most 10. For spoligotype pairs with $H(s_i,s_j)>10$, the genetic proximity is set to zero. As a result, genetic proximity matrix reflects the likelihood of two different pathogens being involved in the same mutation event.

5.3.2.2 Spatial proximity matrix

Given 104 countries, we first define the Country Neighbourhood Matrix (CNM). Given two countries C_i and C_j, the CNM is defined as follows:

$$\text{CNM}(C_i,C_j)=\begin{cases}1, & \text{if } C_i \text{ and } C_j \text{ are neighbours}\\ 0, & \text{otherwise.}\end{cases}$$

Let $L(C_i,C_j)$ be the length of shortest path from C_i to C_j based on Dijkstra's shortest path algorithm on CNM [149]. Then, we define the spatial proximity matrix P_S as follows:

$$P_S(C_i,C_j)=\begin{cases}\dfrac{1}{1+L(C_i,C_j)}, & \text{if } i\neq j,\ L(C_i,C_j)\leq 3\\ 1, & \text{if } i=j\\ 0, & \text{otherwise.}\end{cases}$$

Spatial proximity matrix P_S has values inversely proportional to the length of shortest path between two countries, as long as the shortest-path length is at most 3. For country pairs with shortest-path length $L(C_i,C_j)>3$, the proximity

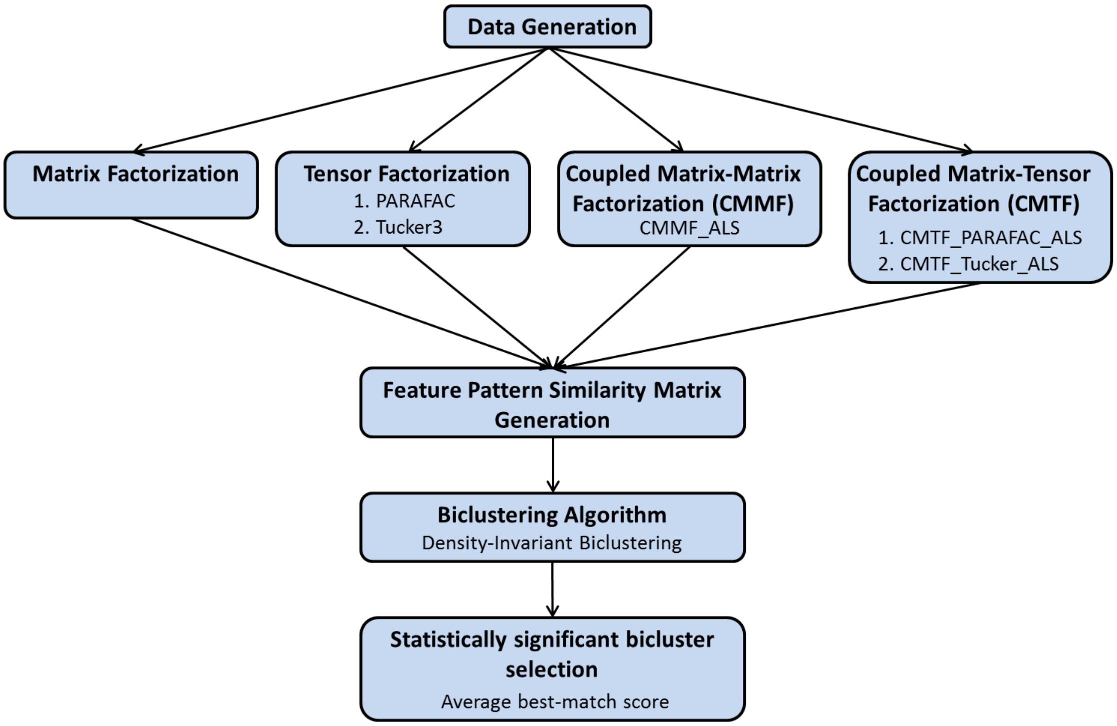

Data Generation
Matrix Factorization
Tensor Factorization
1. PARAFAC
2. Tucker3
Coupled Matrix-Matrix Factorization (CMMF)
CMMF_ALS
Coupled Matrix-Tensor Factorization (CMTF)
1. CMTF_PARAFAC_ALS
2. CMTF_Tucker_ALS
Feature Pattern Similarity Matrix Generation
Biclustering Algorithm
Density-Invariant Biclustering
Statistically significant bicluster selection
Average best-match score

Figure 5.2: Unified Biclustering Framework (UBF). In the first step, the data is generated as a matrix, a tensor, a coupled matrix-matrix, or a coupled matrix-tensor. In the second step, the data in various forms are factorized. In the third step, feature pattern similarity matrix is generated using the factor matrices of the decomposition. In the fourth step, we bicluster data points using density-invariant biclustering algorithm. In the final step, we find the most stable biclusters using average best-match score.

between two countries is set to zero. As a result, spatial proximity matrix reflects the likelihood of patients from two countries being involved in the same transmission event.

5.3.3 UBF: Unified Biclustering Framework

In order to analyze host-pathogen associations using various forms of the raw dataset, we propose the Unified Biclustering Framework (UBF). Based on this framework, we generate the data in the first step, which can be a matrix, a tensor, a coupled matrix-matrix, or a coupled matrix-tensor. In the second step, we decompose the dataset according to its form. In the third step, we generate the feature pattern similarity matrix. In the fourth step, we run the density-invariant biclustering (DIB) algorithm on the feature pattern similarity matrix. Finally, we find statistically significant biclusters and evaluate their biological relevance. Figure 5.2 shows the steps of UBF. The software for UBF is available at http://sourceforge.net/projects/ubf/. Next, we give the details of each step.

5.3.3.1 Data generation

The host-pathogen tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$ can be coupled with genetic proximity matrix $\mathbf{Y} \in \mathbb{R}^{I \times M}$ and spatial proximity matrix $\mathbf{Z} \in \mathbb{R}^{J \times N}$. This flexibility leads to different data configurations which allows simultaneous factorization of different data blocks. Possible data configurations are shown in Figure 5.3. In data configuration 1, the host-pathogen tensor $\underline{\mathbf{X}}$ is summed and contracted along the time mode, and $\hat{\mathbf{X}} \in \mathbb{R}^{I \times J}$ is obtained and used without factorization. In data configuration 2, the original host-pathogen tensor $\underline{\mathbf{X}}$ is used. In data configuration 3, genetic proximity matrix $\mathbf{Y}$ is coupled with the host-pathogen tensor $\underline{\mathbf{X}}$ in the first mode, incorporating the genetic distance into domain knowledge. In data configuration 4, spatial proximity matrix $\mathbf{Z}$ is coupled with the host-pathogen tensor $\underline{\mathbf{X}}$ in the second mode, incorporating the spatial distance into domain knowledge. In data configuration 5, genetic proximity matrix $\mathbf{Y}$ and spatial proximity matrix $\mathbf{Z}$ are coupled with the host-pathogen tensor $\underline{\mathbf{X}}$ in the first and second mode respectively, incorporating both the genetic distance and spatial distance into domain knowledge. In data configuration 6, the host-pathogen tensor $\underline{\mathbf{X}}$ is contracted and summed along the time mode, keeping the genetic proximity matrix $\mathbf{Y}$ and spatial proximity matrix $\mathbf{Z}$ coupled with the contracted host-pathogen tensor, which is now the matrix $\hat{\mathbf{X}}$. We use all six configurations to find biclusters to associate spoligotypes and country of birth of tuberculosis patients, and test the effect of distance measures and time on detected groups.

5.3.3.2 Data factorization

In the second step of UBF, we factorize the dataset according to its form. If the data is a matrix, we use it as is. If it is a tensor, we use tensor decomposition methods, PARAFAC and Tucker3, and find the factor matrices for each mode.

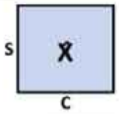
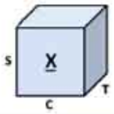
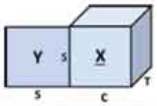
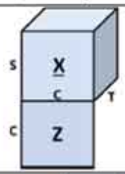
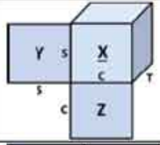
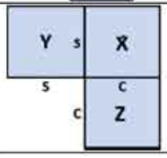
Number	Data configuration	Extra information	Method in UBF
1		—	MBF
2		Time	TBF
3		Time + genetic distance	CMTBF$_g$
4		Time + spatial distance	CMTBF$_s$
5		Time + genetic distance + spatial distance	CMTBF$_{gs}$
6		Genetic distance + spatial distance	CMMBF

Figure 5.3: Data configurations. The mode name S represents spoligotypes, C represents countries, and T represents time in years. The first configuration is a raw *Spoligotypes* × *Countries* matrix decomposed using Matrix Biclustering Framework (MBF) as part of UBF. The second data configuration includes time information as the third mode of the tensor decomposed using Tensor Biclustering Framework (TBF) as part of UBF. Third, fourth and fifth data configurations are the results of concatenating the genetic proximity matrix, spatial proximity matrix, and both respectively, to the host-pathogen tensor. They are decomposed using Coupled Matrix-Tensor Biclustering Framework (CMTBF) as part of UBF. Finally, in data configuration 6, we exclude time information and decompose the resulting data using coupled matrix-matrix biclustering framework (CMMBF) as part of UBF.

When the dataset is a coupled matrix-matrix or matrix-tensor, then we need to simultaneously factorize multiple matrices and/or tensors. We adopt the alternating least squares approach to solve coupled data factorizations. Next, we briefly outline the algorithms we use for coupled matrix-matrix factorization and coupled matrix-tensor factorization.

Coupled matrix-matrix factorization (CMMF): Coupled matrices are simultaneously factorized using the CMMF_ALS algorithm, which we outline next.

CMMF_ALS: The host-pathogen tensor contracted along the time mode becomes the matrix $\hat{\mathbf{X}} \in \mathbb{R}^{I \times J}$. Genetic proximity matrix $\mathbf{Y} \in \mathbb{R}^{I \times I}$ and spatial proximity matrix $\mathbf{Z} \in \mathbb{R}^{J \times J}$ are approximated as in the system of equations (5.1).

$$\begin{aligned} \hat{\mathbf{X}} &\approx \mathbf{A}\mathbf{B}' \\ \mathbf{Y} &\approx \mathbf{A}\mathbf{V}' \\ \mathbf{Z} &\approx \mathbf{B}\mathbf{W}'\,. \end{aligned} \tag{5.1}$$

We want to minimize the following loss function L_1, the sum of Frobenius norm of residuals for each data block:

$$L_1 = ||\hat{\mathbf{X}} - \mathbf{A}\mathbf{B}'||_F^2 + ||\mathbf{Y} - \mathbf{A}\mathbf{V}'||_F^2 + ||\mathbf{Z} - \mathbf{B}\mathbf{W}'||_F^2\,. \tag{5.2}$$

To minimize L_1, we first initialize the factor matrices $\mathbf{A}, \mathbf{B}, \mathbf{V}, \mathbf{W}$ using truncated SVD, and then alternately minimize the loss function by fixing one of them at a time.

$$
\begin{aligned}
&\min_{A,B,V,W} ||\hat{\mathbf{X}} - \mathbf{AB}'||_F^2 + ||\mathbf{Y} - \mathbf{AV}'||_F^2 + ||\mathbf{Z} - \mathbf{BW}'||_F^2 \\
&\min_{A,B,V,W} \mathrm{tr}\left(\left(\hat{\mathbf{X}} - \mathbf{AB}'\right)\left(\hat{\mathbf{X}}' - \mathbf{BA}'\right)\right) + \mathrm{tr}\left((\mathbf{Y} - \mathbf{AV}')(\mathbf{Y}' - \mathbf{VA}')\right) \\
&\qquad + \mathrm{tr}\left((\mathbf{Z} - \mathbf{BW}')(\mathbf{Z}' - \mathbf{WB}')\right) \\
&\min_{A,B,V,W} \mathrm{tr}\left(\hat{\mathbf{X}}\hat{\mathbf{X}}'\right) - 2\mathrm{tr}\left(\mathbf{BA}'\hat{\mathbf{X}}\right) + \mathrm{tr}\left(\mathbf{AB}'\mathbf{BA}'\right) + \mathrm{tr}\left(\mathbf{YY}'\right) - 2\mathrm{tr}\left(\mathbf{VA}'\mathbf{Y}\right) + \\
&\qquad \mathrm{tr}\left(\mathbf{AV}'\mathbf{VA}'\right) + \mathrm{tr}\left(\mathbf{ZZ}'\right) - 2\mathrm{tr}\left(\mathbf{WB}'\mathbf{Z}\right) + \mathrm{tr}\left(\mathbf{BW}'\mathbf{WB}'\right) \\
&\min_{A,B,V,W} - 2\mathrm{tr}\left(\mathbf{BA}'\hat{\mathbf{X}}\right) - 2\mathrm{tr}\left(\mathbf{VA}'\mathbf{Y}\right) - 2\mathrm{tr}\left(\mathbf{WB}'\mathbf{Z}\right) + \mathrm{tr}\left(\mathbf{AB}'\mathbf{BA}'\right) + \mathrm{tr}\left(\mathbf{AV}'\mathbf{VA}'\right) \\
&\qquad + \mathrm{tr}\left(\mathbf{BW}'\mathbf{WB}'\right) \qquad (5.3)
\end{aligned}
$$

Therefore, the objective function (5.3) is:

$$
\begin{aligned}
L = -2\mathrm{tr}\left(\mathbf{BA}'\hat{\mathbf{X}}\right) - 2\mathrm{tr}\left(\mathbf{VA}'\mathbf{Y}\right) - 2\mathrm{tr}\left(\mathbf{WB}'\mathbf{Z}\right) + \mathrm{tr}\left(\mathbf{AB}'\mathbf{BA}'\right) + \mathrm{tr}\left(\mathbf{AV}'\mathbf{VA}'\right) \\
+\mathrm{tr}\left(\mathbf{BW}'\mathbf{WB}'\right) .
\end{aligned} \qquad (5.4)
$$

To minimize the loss function for $\mathbf{A}, \mathbf{B}, \mathbf{V}, \mathbf{W}$ after fixing other factor matrices, we take the derivative of objective function L in equation (5.4), and set it to zero for each factor matrix, which gives the following update rules of matrices in CMMF_ALS:

Update for $\mathbf{A}$:

$$
\begin{aligned}
&\frac{\partial L}{\partial \mathbf{A}} = -2\hat{\mathbf{X}}\mathbf{B} - 2\mathbf{YV} + 2\mathbf{AB}'\mathbf{B} + 2\mathbf{AV}'\mathbf{V} = 0 \\
&\Longrightarrow \mathbf{AB}'\mathbf{B} + \mathbf{AV}'\mathbf{V} = \hat{\mathbf{X}}\mathbf{B} + \mathbf{YV} \\
&\qquad \mathbf{A} = \left(\hat{\mathbf{X}}\mathbf{B} + \mathbf{YV}\right) \backslash \left(\mathbf{B}'\mathbf{B} + \mathbf{V}'\mathbf{V}\right)
\end{aligned}
$$

Update for $\mathbf{B}$:

$$\begin{aligned}\frac{\partial L}{\partial \mathbf{B}} &= -2\hat{\mathbf{X}}'\mathbf{A} - 2\mathbf{Z}\mathbf{W} + 2\mathbf{B}\mathbf{A}'\mathbf{A} + 2\mathbf{B}\mathbf{W}'\mathbf{W} = 0\\ \Longrightarrow & \mathbf{B}\mathbf{A}'\mathbf{A} + \mathbf{B}\mathbf{W}'\mathbf{W} = \hat{\mathbf{X}}'\mathbf{A} + \mathbf{Z}\mathbf{W}\\ \mathbf{B} &= \left(\hat{\mathbf{X}}'\mathbf{A} + \mathbf{Z}\mathbf{W}\right) \setminus (\mathbf{A}'\mathbf{A} + \mathbf{W}'\mathbf{W})\end{aligned}$$

Update for $\mathbf{V}$:

$$\begin{aligned}\frac{\partial L}{\partial \mathbf{V}} &= -2\mathbf{Y}'\mathbf{A} + 2\mathbf{V}\mathbf{A}'\mathbf{A} = 0\\ \Longrightarrow & \mathbf{V}\mathbf{A}'\mathbf{A} = \mathbf{Y}'\mathbf{A}\\ \mathbf{V} &= (\mathbf{Y}'\mathbf{A}) \setminus (\mathbf{A}'\mathbf{A})\end{aligned}$$

Update for $\mathbf{W}$:

$$\begin{aligned}\frac{\partial L}{\partial \mathbf{W}} &= -2\mathbf{Z}'\mathbf{B} + 2\mathbf{W}\mathbf{B}'\mathbf{B} = 0\\ \Longrightarrow & \mathbf{W}\mathbf{B}'\mathbf{B} = \mathbf{Z}'\mathbf{B}\\ \mathbf{W} &= (\mathbf{Z}'\mathbf{B}) \setminus (\mathbf{B}'\mathbf{B})\end{aligned}$$

where $\setminus$ represents right matrix division. The complete CMMF_ALS procedure is summarized in Algorithm 8. In this algorithm, the function `svd_mmf`$(\hat{\mathbf{X}}, \mathbf{Y}, \mathbf{Z})$ initializes the factor matrices $\mathbf{A}, \mathbf{B}, \mathbf{V}, \mathbf{W}$ using truncated SVD with $min(J, M, N)$ components.

Coupled matrix-tensor factorization (CMTF): Coupled matrices and tensors can be simultaneously factorized. For this purpose, we used modifications of PARAFAC and Tucker3 methods. CMTF_PARAFAC_ALS decomposes the tensor using PARAFAC while factorizing the coupled matrices simultaneously. CMTF_PARAFAC_ALS algorithm and its variations exist in the literature. We built another algorithm, extension of Tucker3 to coupled matrix-tensor factorization. CMTF_Tucker_ALS algorithm decomposes the tensor using Tucker3, while simultaneously factorizing the coupled matrices. In the next section, we give the details of these

Algorithm 8 CMMF_ALS($\hat{\mathbf{X}} \in \mathbb{R}^{I\times J}$,$\mathbf{Y} \in \mathbb{R}^{I\times M}$,$\mathbf{Z} \in \mathbb{R}^{J\times N}$)

```
[A, B, V, W] = svd_mmf($\hat{\mathbf{X}}$, $\mathbf{Y}$, $\mathbf{Z}$)
loss(current) = $||\hat{\mathbf{X}} - \mathbf{AB}'||_F^2 + ||\mathbf{Y} - \mathbf{AV}'||_F^2 + ||\mathbf{Z} - \mathbf{BW}'||_F^2$
loss(prev) = loss(current)
$count = 0$
while $((count == 0)\ ||\ (0 < count \leq 10^3\ \&\&\ \frac{|loss(current) - loss(prev)|}{loss(prev)} > 10^{-8}))$ do
    $count++$
    // Solve for A
    $\mathbf{A} = \left(\hat{\mathbf{X}}\mathbf{B} + \mathbf{YV}\right) \setminus (\mathbf{B}'\mathbf{B} + \mathbf{V}'\mathbf{V})$
    // Solve for B
    $\mathbf{B} = \left(\hat{\mathbf{X}}'\mathbf{A} + \mathbf{ZW}\right) \setminus (\mathbf{A}'\mathbf{A} + \mathbf{W}'\mathbf{W})$
    // Solve for V
    $\mathbf{V} = (\mathbf{Y}'\mathbf{A}) \setminus (\mathbf{A}'\mathbf{A})$
    // Solve for W
    $\mathbf{W} = (\mathbf{Z}'\mathbf{B}) \setminus (\mathbf{B}'\mathbf{B})$
    loss(prev) = loss(current)
    loss(current) = $||\hat{\mathbf{X}} - \mathbf{AB}'||_F^2 + ||\mathbf{Y} - \mathbf{AV}'||_F^2 + ||\mathbf{Z} - \mathbf{BW}'||_F^2$
end while
```

algorithms.

CMTF_PARAFAC_ALS: Given the host-pathogen tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I\times J\times K}$ coupled with genetic proximity matrix $\mathbf{Y} \in \mathbb{R}^{I\times I}$ and spatial proximity matrix $\mathbf{Z} \in \mathbb{R}^{J\times J}$, we approximate them as follows:

$$\begin{aligned}
\mathbf{X}_{(1)} &\approx \mathbf{A}\left(\mathbf{C} \odot \mathbf{B}\right)' \\
\mathbf{Y} &\approx \mathbf{AV}' \\
\mathbf{Z} &\approx \mathbf{BW}'
\end{aligned} \tag{5.5}$$

where $\odot$ denotes the Khatri-Rao product. We want to minimize the following loss function which is the sum of squared Frobenius norm of residuals for each data block:

$$L_2 = ||\mathbf{X}_{(1)} - \mathbf{A}\left(\mathbf{C} \odot \mathbf{B}\right)'||_F^2 + ||\mathbf{Y} - \mathbf{AV}'||_F^2 + ||\mathbf{Z} - \mathbf{BW}'||_F^2 . \tag{5.6}$$

CMTF_PARAFAC_ALS is also known as CMTF_ALS algorithm in the literature, which is detailed in earlier studies [150]. Therefore, we skip the details of the algorithm, and only focus on the update step for each factor matrix. Minimization for $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{V}, \mathbf{W}$ alternately returns the following updates at each step of CMTF_PARAFAC_ALS:

Update for **A**:

$$\begin{aligned}
&\min_{A} ||\mathbf{X}_{(1)} - \mathbf{A}\left(\mathbf{C} \odot \mathbf{B}\right)'||_F^2 + ||\mathbf{Y} - \mathbf{A}\mathbf{V}'||_F^2 \\
&\min_{A} ||\underbrace{\left[\mathbf{X}_{(1)}\ \mathbf{Y}\right]}_{T} - \mathbf{A} \underbrace{\left[\left(\mathbf{C} \odot \mathbf{B}\right)'\ \mathbf{V}'\right]}_{K} ||_F^2 \\
&\Longrightarrow \mathbf{A} = \left(\mathbf{T}\mathbf{K}'\right) / \left(\mathbf{K}\mathbf{K}'\right)
\end{aligned}$$

Update for **B**:

$$\begin{aligned}
&\min_{B} ||\mathbf{X}_{(2)} - \mathbf{B}\left(\mathbf{C} \odot \mathbf{A}\right)'||_F^2 + ||\mathbf{Z} - \mathbf{B}\mathbf{W}'||_F^2 \\
&\min_{B} ||\underbrace{\left[\mathbf{X}_{(2)}\ \mathbf{Z}\right]}_{T} - \mathbf{B} \underbrace{\left[\left(\mathbf{C} \odot \mathbf{A}\right)'\ \mathbf{W}'\right]}_{K} ||_F^2 \\
&\Longrightarrow \mathbf{B} = \left(\mathbf{T}\mathbf{K}'\right) / \left(\mathbf{K}\mathbf{K}'\right)
\end{aligned}$$

Update for **C**:

$$\begin{aligned}
&\min_{C} ||\underbrace{\mathbf{X}_{(3)}}_{T} - \mathbf{C} \underbrace{\left(\mathbf{B} \odot \mathbf{A}\right)'}_{K} ||_F^2 \\
&\Longrightarrow \mathbf{C} = \left(\mathbf{T}\mathbf{K}'\right) / \left(\mathbf{K}\mathbf{K}'\right)
\end{aligned}$$

Update for **V**:

$$\begin{aligned}
&\min_{V} ||\mathbf{Y} - \mathbf{A}\mathbf{V}'||_F^2 \\
&\Longrightarrow \mathbf{V} = \left(\left(\mathbf{A}'\mathbf{A}\right) \setminus \left(\mathbf{A}'\mathbf{Y}\right)\right)'
\end{aligned}$$

Update for $\mathbf{W}$:

$$\begin{aligned}&\min_{W}||\mathbf{Z}-\mathbf{B}\mathbf{W}'||_F^2\\ \Longrightarrow &\mathbf{W}=((\mathbf{B}'\mathbf{B})\setminus(\mathbf{B}'\mathbf{Z}))'\end{aligned}$$

CMTF_Tucker_ALS: Next, we extend Tucker3 method to CMTF_Tucker_ALS for coupled matrix-tensor decomposition. This algorithm comes with the flexibility of factorizing the tensor using different number of components for each mode, while simultaneously factorizing the coupled matrices. The host-pathogen tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I\times J\times K}$, genetic proximity matrix $\mathbf{Y} \in \mathbb{R}^{I\times I}$ and spatial proximity matrix $\mathbf{Z} \in \mathbb{R}^{J\times J}$ are approximated as in the system of equations (5.7).

$$\begin{aligned}\mathbf{X}_{(1)} &\approx \mathbf{A}\mathbf{G}_{(1)}\left(\mathbf{C}'\otimes\mathbf{B}'\right)\\ \mathbf{Y} &\approx \mathbf{A}\mathbf{V}'\\ \mathbf{Z} &\approx \mathbf{B}\mathbf{W}'\end{aligned}\tag{5.7}$$

where $\otimes$ denotes the Kronecker product. Note that in the Tucker3 model, the factor matrices $\mathbf{A} \in \mathbb{R}^{I\times P}$, $\mathbf{B} \in \mathbb{R}^{J\times Q}$, $\mathbf{C} \in \mathbb{R}^{K\times R}$ are orthogonal. Then, tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I\times J\times K}$ can be decomposed using a (P,Q,R)-component Tucker3 model, while simultaneously factorizing $\mathbf{Y} \in \mathbb{R}^{I\times I}$ and $\mathbf{Z} \in \mathbb{R}^{J\times J}$ with the factor matrices of the shared mode. We want to minimize the loss function L_3 in Equation (5.8), which is the sum of squared Frobenius norm of residuals for each data block.

$$L_3 = ||\mathbf{X}_{(1)} - \mathbf{A}\mathbf{G}_{(1)}\left(\mathbf{C}'\otimes\mathbf{B}'\right)||_F^2 + ||\mathbf{Y}-\mathbf{A}\mathbf{V}'||_F^2 + ||\mathbf{Z}-\mathbf{B}\mathbf{W}'||_F^2\,. \tag{5.8}$$

To minimize L_3, we first initialize the factor matrices $\mathbf{A}, \mathbf{B}, \mathbf{V}, \mathbf{W}$ using truncated SVD, and then alternately minimize the loss function for one of the variables at a time, while fixing the other variables. The following steps in Equation (5.9) reformulate the minimization of the loss function.

$$\begin{aligned}
&\min_A ||\mathbf{X}_{(1)} - \mathbf{A}\mathbf{G}_{(1)}\left(\mathbf{C}' \otimes \mathbf{B}'\right)||_F^2 + ||\mathbf{Y} - \mathbf{A}\mathbf{V}'||_F^2 \\
&\min_A || \left[\mathbf{X}_{(1)}\ \mathbf{Y}\right] - \left[\mathbf{A}\mathbf{G}_{(1)}\left(\mathbf{C}' \otimes \mathbf{B}'\right)\ \mathbf{A}\mathbf{V}'\right] ||_F^2 \\
&\min_A || \left[\mathbf{X}_{(1)}\ \mathbf{Y}\right] - \left[\mathbf{A}\mathbf{A}'\underbrace{\mathbf{X}_{(1)}\left(\mathbf{C}\mathbf{C}' \otimes \mathbf{B}\mathbf{B}'\right)}_{\mathbf{M}_1}\ \mathbf{A}\mathbf{V}'\right] ||_F^2 \\
&\min_A || \left[\mathbf{X}_{(1)}\ \mathbf{Y}\right] - \left[\mathbf{A}\mathbf{A}'\mathbf{M}_1\ \mathbf{A}\mathbf{V}'\right] ||_F^2 \\
&\min_A + \operatorname{tr}\left(\left(\left[\mathbf{X}_{(1)}\ \mathbf{Y}\right] - \left[\mathbf{A}\mathbf{A}'\mathbf{M}_1\ \mathbf{A}\mathbf{V}'\right]\right)\left(\left[\mathbf{X}_{(1)}\ \mathbf{Y}\right]' - \left[\mathbf{A}\mathbf{A}'\mathbf{M}_1\ \mathbf{A}\mathbf{V}'\right]'\right)\right) \\
&\min_A + \operatorname{tr}\left(\left[\mathbf{X}_{(1)}\ \mathbf{Y}\right]\left[\mathbf{X}_{(1)}\ \mathbf{Y}\right]'\right) - 2\operatorname{tr}\left(\left[\mathbf{X}_{(1)}\ \mathbf{Y}\right]\left[\mathbf{M}_1'\mathbf{A}\mathbf{A}'\ ;\ \mathbf{V}\mathbf{A}'\right]\right) \\
&\qquad + \operatorname{tr}\left(\left[\mathbf{A}\mathbf{A}'\mathbf{M}_1\ \mathbf{A}\mathbf{V}'\right]\left[\mathbf{M}_1\mathbf{A}\mathbf{A}'\ ;\ \mathbf{V}\mathbf{A}'\right]\right) \\
&\min_A - 2\operatorname{tr}\left(\left[\mathbf{X}_{(1)}\ \mathbf{Y}\right]\left[\mathbf{M}_1'\mathbf{A}\mathbf{A}'\ ;\ \mathbf{V}\mathbf{A}'\right]\right) + \operatorname{tr}\left(\mathbf{A}\mathbf{A}'\mathbf{M}_1\mathbf{M}_1'\mathbf{A}\mathbf{A}' + \mathbf{A}\mathbf{V}'\mathbf{V}\mathbf{A}'\right) \\
&\min_A - 2\operatorname{tr}\left(\mathbf{X}_{(1)}\mathbf{M}_1'\mathbf{A}\mathbf{A}' + \mathbf{Y}\mathbf{V}\mathbf{A}'\right) + \operatorname{tr}\left(\mathbf{A}\mathbf{A}'\mathbf{M}_1\mathbf{M}_1'\mathbf{A}\mathbf{A}' + \mathbf{A}\mathbf{V}'\mathbf{V}\mathbf{A}'\right) \\
&\min_A - 2\operatorname{tr}\left(\mathbf{X}_{(1)}\mathbf{M}_1'\mathbf{A}\mathbf{A}'\right) - 2\operatorname{tr}\left(\mathbf{Y}\mathbf{V}\mathbf{A}'\right) + \operatorname{tr}\left(\mathbf{A}\mathbf{A}'\mathbf{M}_1\mathbf{M}_1'\mathbf{A}\mathbf{A}'\right) + \operatorname{tr}\left(\mathbf{A}\mathbf{V}'\mathbf{V}\mathbf{A}'\right) \\
&\min_A - 2\operatorname{tr}\left(\mathbf{M}_1\mathbf{M}_1'\mathbf{A}\mathbf{A}'\right) - 2\operatorname{tr}\left(\mathbf{Y}\mathbf{Y}'\mathbf{A}\mathbf{A}'\right) + \operatorname{tr}\left(\mathbf{A}'\mathbf{M}_1\mathbf{M}_1'\mathbf{A}\right) + \operatorname{tr}\left(\mathbf{A}\mathbf{A}'\mathbf{Y}\mathbf{Y}'\mathbf{A}\mathbf{A}'\right) \\
&\min_A - 2\operatorname{tr}\left(\mathbf{A}'\mathbf{M}_1\mathbf{M}_1'\mathbf{A}\right) - 2\operatorname{tr}\left(\mathbf{A}'\mathbf{Y}\mathbf{Y}'\mathbf{A}\right) + \operatorname{tr}\left(\mathbf{A}'\mathbf{M}_1\mathbf{M}_1'\mathbf{A}\right) + \operatorname{tr}\left(\mathbf{A}'\mathbf{Y}\mathbf{Y}'\mathbf{A}\right) \\
&\min_A - \operatorname{tr}\left(\mathbf{A}'\mathbf{M}_1\mathbf{M}_1'\mathbf{A}\right) - \operatorname{tr}\left(\mathbf{A}'\mathbf{Y}\mathbf{Y}'\mathbf{A}\right) \qquad (5.9) \\
&\text{ s.t. } \mathbf{A}'\mathbf{A} = \mathbf{I}
\end{aligned}$$

where $\mathbf{M}_1 = \mathbf{X}_{(1)}\left(\mathbf{C}\mathbf{C}' \otimes \mathbf{B}\mathbf{B}'\right)$. The Lagrangian of this function is:

$$L_A = -\operatorname{tr}\left(\mathbf{A}'\mathbf{M}_1\mathbf{M}_1'\mathbf{A}\right) - \operatorname{tr}\left(\mathbf{A}'\mathbf{Y}\mathbf{Y}'\mathbf{A}\right) + \operatorname{tr}\left(\lambda\left(\mathbf{A}'\mathbf{A} - \mathbf{I}\right)\right)$$

where λ are the Lagrangian multipliers for the orthogonality constraint $\mathbf{A}'\mathbf{A} = \mathbf{I}$. The derivative of L_A with respect to $\mathbf{A}$ set to zero returns the following equation:

$$\begin{aligned}\frac{\partial L_A}{\partial \mathbf{A}} &= -2\mathbf{M}_1\mathbf{M}_1'\mathbf{A} - 2\mathbf{Y}\mathbf{Y}'\mathbf{A} + \lambda\left(2\mathbf{A}\right) = 0 \\ &\Longrightarrow \left(\mathbf{M}_1\mathbf{M}_1' + \mathbf{Y}\mathbf{Y}'\right)\mathbf{A} = \lambda\mathbf{A} \end{aligned} \tag{5.10}$$

The optimal solution of (5.9) must satisfy Equation (5.10). Thus, $\mathbf{A}$ is composed of first P largest eigenvectors of $(\mathbf{M}_1\mathbf{M}_1' + \mathbf{Y}\mathbf{Y}')$. We denote it as follows:

$$\mathbf{A} = \mathrm{EVD}\left(\mathbf{M}_1\mathbf{M}_1' + \mathbf{Y}\mathbf{Y}', P\right) . \tag{5.11}$$

Similarly, for the second mode, we write the loss function L_3 in Equation (5.8) by matricizing the tensor along the second mode. Then, the objective function is:

$$\begin{aligned} &\min_{B} - \mathrm{tr}\left(\mathbf{B}'\mathbf{M}_2\mathbf{M}_2'\mathbf{B}\right) - \mathrm{tr}\left(\mathbf{B}'\mathbf{Z}\mathbf{Z}'\mathbf{B}\right) \\ &\text{s.t. } \mathbf{B}'\mathbf{B} = \mathbf{I} \end{aligned} \tag{5.12}$$

where $\mathbf{M}_2 = \mathbf{X}_{(2)}\left(\mathbf{C}\mathbf{C}' \otimes \mathbf{A}\mathbf{A}'\right)$. The Lagrangian of this objective function is:

$$L_B = -\mathrm{tr}\left(\mathbf{B}'\mathbf{M}_2\mathbf{M}_2'\mathbf{B}\right) - \mathrm{tr}\left(\mathbf{B}'\mathbf{Z}\mathbf{Z}'\mathbf{B}\right) + \mathrm{tr}\left(\lambda\left(\mathbf{B}'\mathbf{B} - \mathbf{I}\right)\right)$$

where λ are the Lagrangian multipliers for the orthogonality constraint $\mathbf{B}'\mathbf{B} = \mathbf{I}$. The derivative of L_B with respect to $\mathbf{B}$ set to zero returns the following equation:

$$\begin{aligned}\frac{\partial L_B}{\partial \mathbf{B}} &= -2\mathbf{M}_2\mathbf{M}_2'\mathbf{B} - 2\mathbf{Z}\mathbf{Z}'\mathbf{B} + \lambda\left(2\mathbf{B}\right) = 0 \\ &\Longrightarrow \left(\mathbf{M}_2\mathbf{M}_2' + \mathbf{Z}\mathbf{Z}'\right)\mathbf{B} = \lambda\mathbf{B} \end{aligned}$$

which means that $\mathbf{B}$ is composed of first Q largest eigenvectors of $(\mathbf{M}_2\mathbf{M}_2' + \mathbf{Z}\mathbf{Z}')$. We denote it as follows:

$$\mathbf{B} = \mathrm{EVD}\left(\mathbf{M}_2{\mathbf{M}_2}' + \mathbf{Z}\mathbf{Z}', Q\right) . \tag{5.13}$$

For the uncoupled third mode, we write the objective function L_3 in Equation (5.8) by matricizing the tensor along the third mode. The objective function is as follows:

$$\begin{aligned} &\min_{C} - \operatorname{tr}\left(\mathbf{C}'\mathbf{M}_3\mathbf{M}_3'\mathbf{C}\right) \\ &\text{s.t. } \ \mathbf{C}'\mathbf{C} = \mathbf{I} \end{aligned} \tag{5.14}$$

where $\mathbf{M}_3 = \mathbf{X}_{(3)}\left(\mathbf{B}\mathbf{B}' \otimes \mathbf{A}\mathbf{A}'\right)$. The Lagrangian of this function is:

$$L_C = -\operatorname{tr}\left(\mathbf{C}'\mathbf{M}_3\mathbf{M}_3'\mathbf{C}\right) + \lambda\left(\operatorname{tr}\left(\mathbf{C}'\mathbf{C} - \mathbf{I}\right)\right) .$$

The derivative of L_C with respect to $\mathbf{C}$ set to zero returns the following equation:

$$\begin{aligned} &\frac{\partial L_C}{\partial \mathbf{C}} = -2\mathbf{M}_3\mathbf{M}_3'\mathbf{C} + \lambda\left(2\mathbf{C}\right) = 0 \\ &\Longrightarrow \mathbf{M}_3\mathbf{M}_3'\mathbf{C} = \lambda\mathbf{C} \end{aligned}$$

which means that $\mathbf{C}$ is composed of first R largest eigenvectors of $\mathbf{M}_3\mathbf{M}_3'$, or equivalently, first R left singular vectors of $\mathbf{M}_3$. We denote it as follows:

$$\mathbf{C} = \operatorname{SVD}\left(\mathbf{M}_3, R\right) . \tag{5.15}$$

The complete CMTF_Tucker_ALS procedure using these update rules is summarized in Algorithm 9. Note that the function call `hosvd_Tucker(`$\underline{\mathbf{X}}$, $[P, Q, R]$`)` at the beginning of the algorithm initializes factor matrices via truncated SVD using P, Q, R components respectively for each mode. The function `unfoldall(`$\underline{\mathbf{X}}$`)` matricizes the tensor along each mode.

5.3.3.3 Feature pattern similarity matrix generation

We calculate the similarity of feature patterns of a spoligotype s and country c by calculating cosine similarity between feature pattern vectors of them. This is calculated in different ways for different forms of input data. If the input data is a matrix, then the matrix itself is used as the feature pattern similarity matrix

Algorithm 9 CMTF_Tucker_ALS($\underline{\mathbf{X}} \in \mathbb{R}^{I\times J\times K}$, $\mathbf{Y} \in \mathbb{R}^{I\times M}$, $\mathbf{Z} \in \mathbb{R}^{J\times N}$, $[P, Q, R]$)

```
$[\mathbf{A}, \mathbf{B}, \mathbf{C}, \underline{\mathbf{G}}] = \texttt{hosvd\_Tucker}(\underline{\mathbf{X}}, [P, Q, R])$;
$\mathbf{V} = ((\mathbf{A}'\mathbf{A})\backslash(\mathbf{A}'\mathbf{Y}))'$
$\mathbf{W} = ((\mathbf{B}'\mathbf{B})\backslash(\mathbf{B}'\mathbf{Z}))'$
$[\mathbf{X}_{(1)}, \mathbf{X}_{(2)}, \mathbf{X}_{(3)}] = \texttt{unfoldall}(\underline{\mathbf{X}})$
$[\mathbf{G}_{(1)}, \mathbf{G}_{(2)}, \mathbf{G}_{(3)}] = \texttt{unfoldall}(\underline{\mathbf{G}})$
$\text{loss(current)} = ||\mathbf{X}_{(1)} - \mathbf{A}\mathbf{G}_{(1)}(\mathbf{C}' \otimes \mathbf{B}')||_F^2 + ||\mathbf{Y} - \mathbf{A}\mathbf{V}'||_F^2 + ||\mathbf{Z} - \mathbf{B}\mathbf{W}'||_F^2$
$\text{loss(prev)} = \text{loss(current)}$
$count = 0$
while $((count == 0)\ ||\ (0 < count \leq 10^3\ \&\&\ \frac{|loss(current) - loss(prev)|}{loss(prev)} > 10^{-8}))$ do
    $count++$
    // Solve for A
    $\mathbf{M}_1 = \mathbf{X}_{(1)}(\mathbf{C}\mathbf{C}' \otimes \mathbf{B}\mathbf{B}')$
    $\mathbf{A} = \text{EVD}(\mathbf{M}_1\mathbf{M}_1{}' + \mathbf{Y}\mathbf{Y}', P)$
    // Solve for B
    $\mathbf{M}_2 = \mathbf{X}_{(2)}(\mathbf{C}\mathbf{C}' \otimes \mathbf{A}\mathbf{A}')$
    $\mathbf{B} = \text{EVD}(\mathbf{M}_2\mathbf{M}_2{}' + \mathbf{Z}\mathbf{Z}', Q)$
    // Solve for C
    $\mathbf{M}_3 = \mathbf{X}_{(3)}(\mathbf{B}\mathbf{B}' \otimes \mathbf{A}\mathbf{A}')$
    $\mathbf{C} = \text{SVD}(\mathbf{M}_3, R)$
    // Solve for V
    $\mathbf{V} = ((\mathbf{A}'\mathbf{A})\backslash(\mathbf{A}'\mathbf{Y}))'$
    // Solve for W
    $\mathbf{W} = ((\mathbf{B}'\mathbf{B})\backslash(\mathbf{B}'\mathbf{Z}))'$
    $\text{loss(prev)} = \text{loss(current)}$
    $\text{loss(current)} = ||\mathbf{X}_{(1)} - \mathbf{A}\mathbf{G}_{(1)}(\mathbf{C}' \otimes \mathbf{B}')||_F^2 + ||\mathbf{Y} - \mathbf{A}\mathbf{V}'||_F^2 + ||\mathbf{Z} - \mathbf{B}\mathbf{W}'||_F^2$
end while
```

(FPSM). If the data is in tensor form, then FPSM is calculated for PARAFAC as follows. Assume that R-component PARAFAC model on the data matrix returns factor matrix $\mathbf{A} \in \mathbb{R}^{I\times R}$ for the first mode and factor matrix $\mathbf{B} \in \mathbb{R}^{J\times R}$ for the second mode. Then, we first normalize the rows of $\mathbf{A}$ and $\mathbf{B}$, and calculate the feature pattern similarity matrix $FPSM$ as follows:

$$\text{FPSM}_{ij} = \begin{cases} \dfrac{\mathbf{A}_{i.}\ \mathbf{B}'_{j.}}{||\mathbf{A}_{i.}||\ ||\mathbf{B}_{j.}||}, & \text{if } N(i,j) > 0 \\ 0, & \text{otherwise.} \end{cases} \tag{5.16}$$

where $N(i,j)$ represents the number of patients from country j infected with strain i, and $\mathbf{A}_{i.}$ represents the i-th row of $\mathbf{A}$. This calculation is equivalent to cosine similarity of feature vector of i-th sample of $\mathbf{A}$ and feature vector of j-th sample of $\mathbf{B}$, only if there is at least one patient from country j infected with strain i. Calculation of feature pattern matrix after applying Tucker3 model is slightly different. Assume that (P, Q, R)-component Tucker3 model on the data matrix returns factor matrix $\mathbf{A} \in \mathbb{R}^{I \times P}$ for the first mode, factor matrix $\mathbf{B} \in \mathbb{R}^{J \times Q}$ for the second mode, and the core tensor $\underline{\mathbf{G}} \in \mathbb{R}^{P \times Q \times R}$. First, we contract and sum the core tensor $\underline{\mathbf{G}}$ along the third mode and obtain $\hat{\mathbf{G}}$ matrix to calculate the level of interaction between the factors of $\mathbf{A}$ and $\mathbf{B}$. We normalize the rows of $\mathbf{A}\hat{\mathbf{G}}$ and $\mathbf{B}$. Finally, we calculate the feature pattern similarity matrix as the cosine similarity of $\mathbf{A}\hat{\mathbf{G}}$ and $\mathbf{B}$, in Equation (5.17).

$$\begin{aligned} \hat{\mathbf{G}}_{pq} &= \sum_{r=1}^{R} \underline{\mathbf{G}}_{pqr} \\ \mathrm{FPSM}_{ij} &= \begin{cases} \dfrac{\mathbf{A}_{i.}\,\hat{\mathbf{G}}}{||\mathbf{A}_{i.}\,\hat{\mathbf{G}}||}\ \dfrac{\mathbf{B}'_{j.}}{||\mathbf{B}_{j.}||}, & \text{if } N(i,j) > 0 \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \tag{5.17}$$

For coupled factorizations, we use the same equations. After coupled matrix-matrix decomposition, we use Equation (5.16) to find the feature pattern similarity matrix. For coupled matrix-tensor factorization, if CMTF_PARAFAC_ALS is used for factorization, then FPSM is calculated using Equation (5.16). If CMTF_Tucker_ALS is used for factorization, then Equation (5.17) is used to calculate FPSM.

5.3.3.4 Density-invariant biclustering

In this section, we introduce a novel biclustering algorithm based on an existing algorithm and several graph attributes. First, we discretize the input matrix and use it as input to BiMax algorithm to find inclusion-maximal biclusters [60]. Then, we use these biclusters as seed, and find density and variance of these biclusters, which are bicliques. Finally, we find the density-invariant biclusters among candidate inclusion-maximal biclusters.

Given the feature pattern similarity matrix $\mathbf{X} \in \mathbb{R}^{I \times J}$, we use density-invariant biclustering to find coherent biclusters. Let $G = (U, V, E)$ represent a bipartite graph, where U represents the set of genes, or rows in $\mathbf{X}$, V represents the set of conditions, or columns in $\mathbf{X}$, and E represents the weight of the edges connecting vertex set U and vertex set V. The weights E are equivalent to values of matrix $\mathbf{X}$. We want to find biclusters of the following form:

$$B_i = (U_i, V_i, E_i) \tag{5.18}$$

where $B = \bigcup_{i=1}^{n} B_i$ is a biclustering of rows and columns of $\mathbf{X}$. Each bicluster associates a set of rows, in this case spoligotypes, to a set of columns, in this case countries. Notice that each bicluster maps to a submatrix of the original data matrix.

Density-invariant biclustering algorithm first discretizes edge weights using a weight threshold th, and converts the input matrix into a binary matrix $\mathbf{D}$. Then we use the binary inclusion-maximal biclustering algorithm (BiMax) by Prelic et al. on this binary matrix and find a set of candidate biclusters [60]. These biclusters are inclusion-maximal, because the submatrices corresponding to these biclusters are all 1's, and there is no other bicluster which is a superset of it. Output of BiMax algorithm after discretization returns a good starting point for density-invariant biclustering algorithm. Next, we focus on these candidate biclusters. For this purpose, we define the density and variance of a graph.

Definition 1. ***Density of a graph:*** *Density of a graph is the average weight of its edges. Given a graph $G = (V, E)$ where w(e) represents the weight of edge e $\in E$, the density of graph G is calculated as follows:*

$$d(G) = \frac{\sum_{e \in E} w(e)}{\binom{|V|}{2}}.$$

Definition 2. ***Variance of a graph:*** *Variance of a graph is the standard deviation of its edge weights. Given a graph $G = (V, E)$ where w(e) represents the weight of* *edge $e \in E$, the variance of graph G is calculated as follows:*

$$v(G) = \sqrt{\frac{1}{|E|-1} \sum_{e \in E} \left(w\left(e\right) - \bar{w}\right)^2}\ .$$

Using the density and variance of graphs, we can define a new set of graphs which are bounded by their edge weights. Next, we define the α-dense β-variant biclusters, or density-invariant biclusters, which are graphs of the form $B = (U, V, E)$ with density $d(B) \geq \alpha$ and variance $v(B) \leq \beta$, and similarly for all one-vertex-induced subgraphs of $B = (U, V, E)$.

Definition 3. ***Density-invariant bicluster:*** *Let $B = (U, V, E)$ be a bicluster, where edges in E connect vertices in U to vertices in V. Bicluster B is an α-dense bicluster if $d(B) \geq \alpha$, and it is a β-variant bicluster if $v(B) \leq \beta$. Define B' as an induced subgraph of B after removing one vertex, either from vertex set U or vertex set V. Bicluster $B = (U, V, E)$ is an (α, β)-density-invariant bicluster, or density-invariant bicluster, if B and all its one-vertex-induced subgraphs are α-dense β-variant. In short, bicluster B is a density-invariant bicluster if the following conditions hold:*

1. $d(B) \geq \alpha$, $v(B) \leq \beta$

2. $d(B') \geq \alpha$, $v(B') \leq \beta \quad \forall B' = B \setminus \{m\}$ *where* $m \in U \cup V$, $|B'| > 0$.

Notice that a density-invariant bicluster forms a biclique with average weight bounded from below, and variance of weights bounded from above. All induced subgraphs obtained after removing one vertex from a density-invariant bicluster are still α-dense and β-variant, but not necessarily density-invariant biclusters. At this point, we define strong antimonotonicity of a graph, which was introduced in Pao et al. [61].

Definition 4. ***Strong antimonotonicity:*** *A graph attribute is strong antimonotone if for each graph $G = (V, E)$ with the attribute, every induced subgraph $G' = G - \{v\}$ has the attribute, where $v \in V$.*

According to the definition of strong antimonotonicity, the attribute of being a density-invariant graph or bicluster is not strongly antimonotone. This is because the vertex-induced subgraphs of the original graph are α-dense and β-variant, but their vertex-induced subgraphs need not be (α, β)-density-invariant biclusters.

Finally, we iterate over candidate biclusters found as output from BiMax algorithm and find density-invariant biclusters among these candidate biclusters. This results in strongly connected and more homogeneous biclusters. Algorithm 10 summarizes `DensityInvariantBiclustering` procedure.

Algorithm 10 `Biclusters` = `DensityInvariantBiclustering`($\mathbf{X} \in \mathbb{R}^{I\times J}$, th, α, β)

Input: Data matrix $\mathbf{X} \in \mathbb{R}^{I\times J}$, discretization threshold th, density threshold α, variance threshold β.

Output: Density-invariant biclusters `Biclusters`.

```
D = discretize(X, th)
CandidateBiclusters = BiMax(D)
Biclusters = ∅
for i=1:1:length(CandidateBiclusters) do
    $B(U, V, E)$ = CandidateBiclusters(i)
    $check1 = (d(B) \geq \alpha)$ && $(v(B) \leq \beta)$
    $check2 = true$
    M = $U \cup V$
    for j=1:1:length(M) do
        $m = M(j)$
        $B' = B \setminus \{m\}$
        if $((B' \neq \emptyset)$ && $!(d(B') \geq \alpha$ && $v(B') \leq \beta))$ then
            $check2 = false$
            break
        end if
    end for
    if ($check1$ && $check2$) then
        Biclusters = Biclusters $\cup \{B\}$
    end if
end for
```

In `DensityInvariantBiclustering` algorithm, `discretize`$(\mathbf{X}, th)$ function discretizes the input data matrix as follows:

$$\mathbf{D}_{ij} = \begin{cases} 1, & \text{if } \mathbf{X}_{ij} \geq th \\ 0, & \text{otherwise.} \end{cases}$$

`BiMax` algorithm is run on this binary matrix $\mathbf{D}$, and inclusion-maximal biclusters are obtained. Then, among these candidate biclusters, density-invariant biclusters are found.

5.3.3.5 Statistically significant bicluster selection

In order to find statistically significant biclusters, we sample 90% of the patients, and rerun the biclustering algorithm, and obtain 20 new biclusterings. Then, we calculate the stability of each density-invariant bicluster found in the previous step using average best-match score. First, we calculate the match score of two biclusters $B_1 = (G_1, C_1)$, $B_2 = (G_2, C_2)$, where G_1, G_2 represent gene sets and C_1, C_2 represent condition sets. Similar to Prelic et al. and Lie et al. [60, 151], the match score of biclusters $B_1 = (G_1, C_1)$, $B_2 = (G_2, C_2)$ is calculated as follows:

$$\texttt{match}(B_1, B_2) = \frac{|G_1 \cap G_2| + |C_1 \cap C_2|}{|G_1 \cup G_2| + |C_1 \cup C_2|}\,. \tag{5.19}$$

Let $M = \bigcup_{i=1}^{k} B_i^*$ be a biclustering of the subsample of the dataset. We compare a bicluster $B = (G, C)$ to all biclusters in $B_i^* \in M$, and assign the maximum match value as the best-match score:

$$\texttt{best\_match}(B, M = \bigcup_{i=1}^{k} B_i^*) = \max_{B_i^* \in M} \texttt{match}(B, B_i^*)\,. \tag{5.20}$$

Finally, we take the average best-match score of each bicluster B by comparing them to each biclustering M_i, and obtain the average best-match score of bicluster B as follows:

$$\texttt{average\_best\_match}(B, \bigcup_{i=1}^{n} M_i) = \frac{\sum_{i=1}^{n} \texttt{best\_match}(B, M_i)}{n}\,. \tag{5.21}$$

We pick the biclusters with ≥ 95% average best-match score as statistically significant biclusters, and evaluate their biological relevance. If there are no significant biclusters, we report top 5 stable biclusters with their average best-match scores.

Table 5.1: Biclustering results for each data configuration, including density-invariant biclustering algorithm parameters and number of density-invariant biclusters (DIB). For TBF, PARAFAC and Tucker3 model, results are listed separately. Similarly, for CMTBF, CMTF_PARAFAC_ALS and CMTF_Tucker_ALS, results are listed separately. When there are no stable biclusters with average best-match score $\geq$ 95%, five most stable biclusters are picked as the stable biclusters.

# Configuration	Method	DIB parameters (th, α, β)	# DIB
1	MBF	0.80, 0.80, 0.15	8
2	TBF (PARAFAC)	0.98, 0.89, 0.01	170
	TBF (Tucker3)	0.60, 0.60, 0.40	5
3	CMTBF$_g$ (CMTF_PARAFAC_ALS)	0.60, 0.60, 0.40	0
	CMTBF$_g$ (CMTF_Tucker_ALS)	0.70, 0.70, 0.30	4
4	CMTBF$_s$ (CMTF_PARAFAC_ALS)	0.80, 0.90, 0.10	21
	CMTBF$_s$ (CMTF_Tucker_ALS)	0.80, 0.85, 0.15	6
5	CMTBF$_{gs}$ (CMTF_PARAFAC_ALS)	0.98, 0.99, 0.01	0
	CMTBF$_{gs}$ (CMTF_Tucker_ALS)	0.60, 0.70,0.30	5
6	CMMBF	0.60, 0.60, 0.40	17

5.4 Results

In order to find host-pathogen associations in tuberculosis patient dataset, we biclustered spoligotypes and countries using six different data configurations shown in Figure 5.3. For each data configuration, we followed the steps of Unified Biclustering Framework (UBF), and found the most stable biclusters. Table 5.1 shows the parameters of `DensityInvariantBiclustering` (th, α, β) and number of density-invariant biclusters for each data configuration. Note that PARAFAC and Tucker3 variants of TBF, CMTF_PARAFAC_ALS and CMTF_Tucker_ALS variants of CMTBF are listed separately. Next, we evaluate the statistical significance and biological relevance of biclusters for each data configuration, and find host-pathogen associations within the whole patient dataset and within each major lineage.

5.4.1 Biclusters using spoligotypes and country of birth

We first contract and sum the host-pathogen tensor along the time mode and find biclusters based on the distribution of spoligotypes to countries of birth, as in data configuration 1 in Figure 5.3. In this setting, no distance measure or time is added to the domain knowledge. Table 5.2 shows the density-invariant biclusters. Bicluster B1 suggests that patients from Haiti are infected with ST1162

Table 5.2: Biclustering results on data configuration 1 using UBF. Biclusters associate spoligotypes to country of birth of patients. For spoligotypes, SIT number, major lineage based on CBN, and sublineage based on KBBN are listed. For countries, the name and the TB continent are listed. Bicluster B16 represents the well-known association between patients from Philippines and EAI2-Manila strains.

Bicluster	**Number of patients**	**Spoligotypes**			**Countries**	
		SIT no	**Major lineage**	**Sublineage**	**Name**	**TB continent**
B11	5	ST1162	East-Asian	Beijing	Haiti	Americas
		ST398	Euro-American	LAM4		
B13	19	ST265	East-Asian	Beijing	China	East Asia
		ST422	*M. bovis*	BOV_1		
		ST89	Indo-Oceanic	EAI5		
		ST287	Indo-Oceanic	EAI2-Manila		
		ST1268	Euro-American	T5		
		ST25	East-African Indian	CAS1-Delhi		
		ST732	Euro-American	T1		
B14	6	ST1908	Euro-American	H3	Ecuador	Americas
		ST58	Euro-American	T5		
B15	6	ST43	Indo-Oceanic	EAI6-BGD1	Dominican Republic	Americas
		ST848	Euro-American	T2		
		ST511	Euro-American	H3		
B16	2	ST897	Indo-Oceanic	EAI2-Manila	Philippines	Southeast Asia
B17	2	ST447	Euro-American	T1	Bangladesh	Indian Subcontinent
B18	4	UST251	Euro-American	S	Mexico	Americas
		ST1154	Euro-American	LAM9		

strain, a Beijing strain, and ST398, a LAM4 strain. Bicluster B12 is listed in the supplementary material due to its size: http://tbinsight.cs.rpi.edu/UBFsupp.rar. This bicluster contains 848 patients from United States who are infected with 63 different strains. One of these strains is the transmissive Beijing strain ST1 which initiated many outbreaks in United States [152, 153]. Bicluster B13 shows that patients from China are infected with 7 different strains. Bicluster B14 shows that ST1908 and ST58 are two Ecuadorian isolates belonging to Euro-American lineage. Bicluster B16 is a well-known association, and suggests that patients from Philippines are infected with an EAI2-Manila strain, ST897. Bicluster B18 suggests that Mexican patients, as neighbours of United States, are infected with UST251 and ST1154, two Euro-American strains. The five most stable biclusters are B16, B17, B18, B11, B14, and their average best-match scores are in the range [0.1667, 0.2]. One may argue that biclusters with few patients does not constitute a strong host-pathogen association. This suggests that TB detection rate should be increased to gather more patient data and make more accurate inferences on host-pathogen association.

5.4.2 Incorporating time

The original host-pathogen tensor has time as the third mode. Therefore, when we found biclusters using the host-pathogen tensor as in data configuration 2 of Figure 5.3, we account for distribution of spoligotypes to countries of birth through time, in this case years from 2001 to 2007. When we use PARAFAC to decompose the host-pathogen tensor, we found 170 density-invariant biclusters. Here, we focus on five most stable biclusters when PARAFAC model is used. Average best-match scores of these five biclusters range from 0.6915 to 0.7295. The full list of these biclusters can be found in the supplementary material. Bicluster B211 associates Vietnamese patients to 11 strains belonging to Euro-American, East Asian, Indo-Oceanic and East-African Indian lineages. Bicluster B212 suggests that patients from Peru are infected with 17 different strains belonging to Euro-American, Indo-Oceanic, and East Asian lineages. Bicluster B214 is shown in Table 5.3. There are 111 patients in bicluster B214 from India, Peru and Vietnam, which are infected with 6 Euro-American strains and one East Asian strain. Notice that this East Asian strain is ST1, which is the transmissive Beijing strain. This suggests that some of the patients in this bicluster must be involved in the outbreaks in United States initiated by ST1 Beijing strains.

When Tucker3 model is used to decompose the host-pathogen tensor, we find 5 density-invariant biclusters. Their average-best match scores range from 0.04 to 0.18, which shows that biclusters found using Tucker3 model are less stable compared to the ones found using PARAFAC model. These five biclusters, bicluster B221 to B225, are listed in the supplementary material. Bicluster B221 suggests that US patients are infected with 31 different strains, and bicluster B222 suggests that Chinese patients are infected with 11 strains belonging to Euro-American, East-African Indian, and *M. bovis* lineages. Note that no East Asian strain is associated with Chinese patients, which suggests that our biclustering analysis on the host-pathogen tensor is introducing noise when time is added into domain knowledge. Bicluster B225, also listed in Table 5.3, has 4 patients and suggests that Mexican patients are infected with ST294, ST290 and ST176 strains, all members of Euro-American lineage.

Table 5.3: Biclustering results on data configuration 2 using PARAFAC and Tucker3 models via UBF on the host-pathogen tensor. Bicluster B214 associates patients from India, Peru and Vietnam to 6 Euro-American strains and the transmissive East Asian Beijing strain ST1. Bicluster B224 groups Mexican patients infected with three different Euro-American strains.

Bicluster	**Number of patients**	**Spoligotypes**			**Countries**	
		SIT no	**Major lineage**	**Sublineage**	**Name**	**TB continent**
B214	111	ST53	Euro-American	T1	India	Indian Subcontinent
		ST17	Euro-American	LAM2	Peru	Americas
		ST1	East Asian	Beijing	Vietnam	Southeast Asia
		ST197	Euro-American	X3		
		ST61	Euro-American	LAM10-CAM		
		ST119	Euro-American	X1		
		ST42	Euro-American	LAM9		
B225	4	ST294	Euro-American	H3	Mexico	Americas
		ST290	Euro-American	LAM9		
		ST176	Euro-American	LAM6		

5.4.3 Incorporating time and distance measures

Next, we concatenate distance matrices one at a time, and finally both of them, to the host-pathogen tensor. Concatenation of genetic proximity matrix results in data configuration 3, concatenation of spatial proximity matrix results in data configuration 4, and concatenation of both matrices results in data configuration 5. We factorize these matrices using coupled matrix-tensor factorization via CMTF_PARAFAC_ALS and CMTF_Tucker_ALS, and report statistically significant and biologically relevant biclusters. The full list of biclusters can be found in the supplementary material.

If we use genetic distance matrix only, factorization via CMTF_PARAFAC_ALS results in no density-invariant biclusters. When the coupled matrix-tensor is decomposed via CMTF_Tucker_ALS, 4 stable clusters are found, the stability of which range from 0.08 to 0.19. Two of these biclusters, B321 and B323, are listed in Table 5.4. Bicluster B321 groups 32 patients from Ecuador, infected with ST53, ST62, ST51, ST1908, which are all Euro-American strains. Notice that ST53 and ST51 belong to T1 sublineage, which is a class of ill-defined Euro-American strains. Bicluster B323 contains 4 patients from Mexico, all infected with ST52, a Euro-American T2 strain.

If we use spatial distance matrix only, factorization via CMTF_PARAFAC_ALS results in 21 density-invariant biclusters, and we picked 5 most stable biclusters among them. The stability values of these biclusters range from 0.26 to 0.32. Table

Table 5.4: Biclustering results on data configuration 3, 4, 5 using CMTF_PARAFAC_ALS and CMTF_Tucker_ALS algorithms via UBF on the coupled matrix-tensor. Biclusters B411 and B412 suggests that Euro-American strains ST908 and ST904 infects patients from four spatially close countries in Americas respectively. Bicluster B421 suggests that transmissive Beijing strain ST1 is wide-spread and infects patients from three different TB continents. Bicluster B422 groups patients from two neighbour countries, Malaysia and Philippines, who are infected with Beijing strain ST1 and X2 strain ST38.

Bicluster	Number of patients	Spoligotypes			Countries	
		SIT no	**Major lineage**	**Sublineage**	**Name**	**TB continent**
B321	32	ST53 ST62 ST51 ST1908	Euro-American Euro-American Euro-American Euro-American	T1 H1 T1 H3	Ecuador	Americas
B323	4	ST52	Euro-American	T1	Mexico	Americas
B411	6	ST908	Euro-American	LAM2	Dominican Rep. Puerto Rico Trinidad and Tobago United States	Americas Americas Americas Americas
B412	6	ST904	Euro-American	T5	Ecuador Haiti Trinidad and Tobago United States	Americas Americas Americas Americas
B414	6	ST904 ST908	Euro-American Euro-American	T5 LAM2	Trinidad and Tobago United States	Americas Americas
B421	32	ST1	East Asian	Beijing	Taiwan Barbados Dominica Malaysia Myanmar Philippines	East Asia Americas Americas Southeast Asia Southeast Asia Southeast Asia
B422	27	ST1 ST38	East Asian Euro-American	Beijing X2	Malaysia Philippines	Southeast Asia Americas
B425	2	ST93	Euro-American	LAM5	Honduras	Americas
B525	11	ST167 ST42 ST57 ST904 ST187 ST1867	Euro-American Euro-American Euro-American Euro-American *M. africanum* *M. africanum*	T1 LAM9 LAM10-CAM T5 AFRI_1 AFRI_1	Haiti	Americas

5.4 shows 3 of these biclusters. Bicluster B411 suggests that Euro-American LAM2 strain ST908 infects patients from Dominican Republic, Puerto Rico, Trinidad Tobago, and Unites States, all from Americas. Notice how geographically close countries are collected together in a bicluster in the host-pathogen association analysis by incorporating spatial proximity into domain knowledge. Bicluster B412 suggests that Euro-American T5 strain ST904 infects patients from Ecuador, Haiti, Trinidad Tobago, and United States, which are again all in Americas. Bicluster B414 includes strains of both Bicluster B411 and B412, and combines the two common countries in these biclusters. It suggests that Euro-American T5 strain ST904 and Euro-American LAM2 strain ST908 infect patients from Trinidad Tobago and United States. When we factorize the coupled matrix-tensor via CMTF_Tucker_ALS in

UBF, 6 density-invariant biclusters are found, and we picked 5 most stable biclusters among them, with average best-match score ranging from 0.09 to 0.50. Table 5.4 shows 3 of these biclusters. Bicluster B421 points out that transmissive ST1 Beijing strain is wide-spread, and it infects patients from Taiwan, Barbados, Dominica, Malaysia, Myanmar, and Philippines, which cover 3 different TB continents: East Asia, Americas, and Southeast Asia. This shows that, even if we use spatial proximity matrix to narrow down transmission events, transmissive ST1 strain is still associated with patients from multiple TB continents. Bicluster B422 contains 27 patients from Philippines and Malaysia, both from Southeast Asia, which are infected with ST1 and ST38 strains. Notice how these countries are grouped together using the spatial proximity matrix. Bicluster B425 consists of 2 patients from Honduras, both infected with Euro-American LAM5 strain ST93.

If we concatenate both genetic and spatial proximity matrices, factorization via CMTF_PARAFAC_ALS does not assign any density-invariant biclusters. When the coupled matrix-tensor is decomposed via CMTF_Tucker_ALS, we find 5 density-invariant biclusters, with average best-match score values ranging from 0.11 to 0.30. Table 5.4 shows one of these biclusters. Bicluster B525 contains 11 patients from Haiti which are infected with Euro-American strains ST167, ST42, ST57, ST904, and *M. africanum* AFRI_1 strains ST187 and ST1867. The full list of biclusters can be found in supplementary material. Notice that there is no order in stability of biclusters found using CMTF_PARAFAC_ALS and CMTF_Tucker_ALS. However, biclusters found using CMTF_Tucker_ALS are more biologically coherent. This shows that that high stability does not imply biological relevance.

5.4.4 Incorporating distance, but not time

Finally, in the last data configuration, we use genetic distance, spatial distance, but not time. This reduces the mutation path length and transmission path length, which increases the likelihood of mutation between the set of strains and transmission between the set of patients. To do so, we contract and sum the host-pathogen tensor along the time mode, and concatenate the genetic proximity matrix and spatial proximity matrix. We bicluster spoligotypes and countries using CMMF_ALS

Table 5.5: Biclustering results on data configuration 6 using CMMF_ALS via UBF on the coupled matrix-matrix. Bicluster B64 groups patients from Bangladesh who are infected with two strains of ill-defined sublineages: Indo-Oceanic EAI5 strain ST1391 and Euro-American T1 strain ST58.

Bicluster	Number of patients	Spoligotypes			Countries	
		SIT no	**Major lineage**	**Sublineage**	**Name**	**TB continent**
B64	3	ST1391	Indo-Oceanic	EAI5	Bangladesh	Indian Subcontinent
		ST58	Euro-American	T1		
B66	19	ST1162	East Asian	Beijing	Haiti	Americas
		ST168	Euro-American	H3		
		ST398	Euro-American	LAM4		
		ST57	Euro-American	LAM10-CAM		
		ST874	Euro-American	S		
		UST256	Euro-American	H1		
		ST541	East Asian	Beijing		
		ST1867	*M. africanum*	AFRI_1		
		ST822	Euro-American	LAM9		
		ST546	Euro-American	X3		
		ST3	Euro-American	LAM2		

on this dataset in UBF. There are 17 density-invariant biclusters, and we picked the ones with average best-match score of 90% and above. Full list of these biclusters are in the supplementary material. Table 5.5 shows two of these biclusters, B64 and B66. Bicluster B64 contains 3 patients from Bangladesh infected with Indo-Oceanic EAI5 strain ST1391 and Euro-American T1 strain ST447. Notice that EAI5 is a generic sublineage of Indo-Oceanic lineage, and T1 is a generic sublineage of Euro-American lineage, and they are both ill-defined. Bicluster B66 contains 19 patients from Haiti infected with Euro-American, East Asian and *M. africanum* strains. Haiti is an island next to Dominican Republic and immigrants of Haiti must have brought strains belonging to various lineages.

5.4.5 Host-pathogen association within each major lineage

The six phylogeographic major lineages determined by CBN are established. Therefore, we subdivide the patient dataset based on six major lineages, and run UBF on each of them. We used data configuration 6, since it resulted in both stable and biologically relevant biclusters in the complete patient dataset. We found the most stable host-pathogen associations for each major lineage and reported their biological relevance.

Table 5.6 shows some of the most stable and biologically relevant biclusters. The full list of biclusters can be found in the supplementary material. Bicluster B711 of Euro-American lineage in the list of supplementary material contains 628 US pa-

Table 5.6: Biclustering results on data configuration 6 using CMMF_ALS via UBF on the coupled matrix-matrix for each major lineage. Bicluster B712 suggests that Mexican patients are likely to be infected with UST251, ST478, and ST1154 strains, given that the pathogen is a Euro-American strain. Bicluster B742 groups 212 US patients and shows that US patients are commonly infected with Beijing strains, including the transmissive ST1 strain. 291 patients in bicluster B743 shows that Beijing strains ST260, ST265 and the transmissive ST1 strain infects both Chinese and US patients. Biclusters B761 and B762 suggest that, given that MTBC is an *M. bovis* strain, it is more likely to infect a patient from Dominican Republic if it is a BOV or BOV_1 strain, and more likely to infect a US patient if it is a BOV_2 strain.

Bicluster	Number of patients	Spoligotypes			Countries	
		SIT no	Major lineage	Sublineage	Name	TB continent
B712	5	UST251	Euro-American	S	Mexico	Americas
		ST478	Euro-American	X2		
		ST1154	Euro-American	LAM9		
B732	9	ST471	East-African Indian	CAS1-Delhi	China	East Asia
		ST25	East-African Indian	CAS1-Delhi		
		ST381	East-African Indian	CAS1-Delhi		
		ST21	East-African Indian	CAS		
		ST203	East-African Indian	CAS		
		UST167	East-African Indian	EAI5		
B733	11	ST381	East-African Indian	CAS1-Delhi	China Dominican Republic	East Asia Americas
		ST25	East-African Indian	CAS1-Delhi		
		ST21	East-African Indian	CAS		
		UST167	East-African Indian	EAI5		
B741	7	ST1162	East Asian	Beijing	Haiti	Americas
		ST941	East Asian	Beijing		
		ST541	East Asian	Beijing		
		ST1168	East Asian	Beijing		
B742	212	UST1	East Asian	Beijing	United States	Americas
		ST255	East Asian	Beijing		
		ST260	East Asian	Beijing		
		ST941	East Asian	Beijing		
		ST265	East Asian	Beijing		
		ST190	East Asian	Beijing		
		ST1	East Asian	Beijing		
B743	291	ST260	East Asian	Beijing	China United States	East Asia Americas
		ST265	East Asian	Beijing		
		ST1	East Asian	Beijing		
B751	17	ST325	*M. africanum*	AFRI_1	United States	Americas
		ST326	*M. africanum*	AFRI_1		
		ST187	*M. africanum*	AFRI_1		
		ST181	*M. africanum*	AFRI_1		
		ST319	*M. africanum*	AFRI_2		
		ST331	*M. africanum*	AFRI_2		
		UST229	*M. africanum*	AFRI_2		
B761	3	ST479	*M. bovis*	BOV	Dominican Republic	Americas
		ST481	*M. bovis*	BOV_1		
B762	9	ST409	*M. bovis*	BOV_2	United States	Americas
		ST683	*M. bovis*	BOV_2		

tients infected with 61 different strains. Bicluster B712 listed in Table 5.6 suggests a strong association between Mexican patients and pathogens of three Euro-American strains, S strain UST251, X2 strain ST478, and LAM9 strain ST1154. Notice that all strains belong to different sublineages of Euro-American lineage. The average best-match score of this bicluster is 0.7783. Bicluster B721 of Indo-Oceanic lineage listed in the supplementary material suggests an association between 40 Chinese patients and 16 different Indo-Oceanic strains, belonging to various sublineages. The stability value of 0.9621 suggests that this is a strong host-pathogen association.

Bicluster B732 listed in Table 5.6 contains 9 Chinese patients infected with CAS1-Delhi, CAS and EAI5 strains of East-African Indian lineage. Similarly, bicluster B733 suggests that patients from China and Dominican Republic are likely to be infected with the following East-African Indian strains: CAS1-Delhi strains ST381 and ST25, CAS strain ST21, and EAI5 strain UST167. Bicluster B741 suggests that Haitian patients are infected with the following Beijing strains: ST1162, ST941, ST541, ST1168. Similarly, 212 US patients in bicluster B742 suggests that US patients are infected commonly with the following Beijing strains: UST1, ST255, ST260, ST941, ST265, ST190, and the transmissive ST1 strain. 291 patients in bicluster B743 suggest that both Chinese and US patients are infected with the following Beijing strains very frequently: ST260, ST265, and the transmissive ST1 strain. This shows that Beijing strains brought to the US by Chinese immigrants infect both Chinese and US patients in the US.

Bicluster B751 shows that US patients are infected with AFRI_1 strains ST325, ST326, ST187, ST181 and AFRI_2 strains ST319, ST331, UST229 of *M. africanum* lineage. Bicluster B761 suggests that patients from Dominican Republic are likely to be infected with BOV strain ST479 and BOV_1 strain ST481 belonging to *M. bovis* lineage. On the other hand, bicluster B762 suggests that US patients are infected BOV_2 strains ST409 and ST683 belonging to *M. bovis* lineage. These two biclusters suggest that, given an *M. bovis* strain, it is likely to infect a patient from Dominican Republic if it is a BOV or BOV_1 strain, whereas it is more likely to have infected a US patient if the strain is a BOV_2 strain.

5.5 Discussion and Conclusion

We developed the Unified Biclustering Framework (UBF) to find host-pathogen associations in tuberculosis patients. To our knowledge, this is the first study to restate host-pathogen association analysis as a biclustering problem. UBF is flexible in the sense that distance and time can be added into domain knowledge of data analysis via coupled matrix-matrix and matrix-tensor factorization. This enables genome-phenome data fusion in one unsupervised learning framework.

Each bicluster refers to a possible host-pathogen association. We found statistically significant biclusters, some of which represent well-known host-pathogen relationships and some of which reveal new associations. For instance, bicluster B16 shows the well-known association of patients from Philippines and EAI2-Manila strains. Similarly, biclusters B742 and B743 shows that many US patients are infected with Beijing strains including ST1 strain, a well-known initiator of many outbreaks in the US. On the other hand, we also found new patient-strain relationships via genome-phenome data fusion by adding genetic proximity, spatial proximity and time into domain knowledge. For instance, bicluster B422 groups patients from two neighbour countries, Malaysia and Philippines, who are infected with Beijing strain ST1 and X2 strain ST38. Biclusters B761 and B762 suggest that patients from Dominican Republic are infected with BOV and BOV_1 strains of *M. bovis* lineage, whereas US patients are infected with BOV_2 strains of *M. bovis* lineage. Note that although we picked statistically significant biclusters, statistical significance does not imply biological relevance [154]. However, these new stable biclusters lead to new host-pathogen associations.

Host-pathogen association analysis can be extended by adding new patient and strain attributes. As future work, we will add MIRU and RFLP, two biomarkers of MTBC, into this analysis. In addition, we will add other patient attributes such as age group, ethnicity, homelessness and other risk factors of TB. We will also speed up UBF using line search in ALS-based coupled factorization algorithms. This will enhance both the speed and accuracy of coupled factorizations, which will lead to more accurate host-pathogen associations.

CHAPTER 6
CONCLUSIONS AND FUTURE WORK

This thesis made three contributions to algorithmic data fusion methods in order to utilize multiple sources of information from MTBC strains and TB patients. In the first one, we used multiple biomarkers of MTBC in one clustering framework and subdivided major lineages into sublineages. Next, we used multiple biomarkers of MTBC to examine the evolution of spoligotypes. Finally, we combined genomic data from MTBC strains and phenomic data from TB patients via one biclustering framework, and detected host-pathogen associations.

First, we subdivided major lineages of MTBC into sublineages using the Tensor Clustering Framework (TCF) on multiple-biomarker tensors (MBT). The multiple-biomarker tensor holds data from two biomarkers, spoligotypes and MIRU patterns. We factorize the multiple-biomarker tensor into its components using multiway models. We use the factor matrix for strain mode as input to our improved k-means algorithm. Then, we cluster MTBC strains into sublineages. Our new definition of sublineages based on two biomarkers confirm some of the existing sublineages, and suggests subdividing or merging other sublineages.

Second, we built a new mutation model for spoligotypes based on two biomarkers of MTBC, spoligotypes themselves and MIRU patterns. The model uses a maximum parsimony method based on three genetic distance measures on two biomarkers. The resulting spoligoforest shows the mutation history of spoligotypes. Based on the topology of the spoligoforest, number of descendant spoligotypes follows a power-law distribution. In addition, number of mutations at each spacer in the DR region follows a spatially bimodal distribution. Based on this observation, we built two alternative models for mutation length frequency: Starting Point Model (SPM) and Longest Block Model (LBM). Both models plausibly fit mutation length frequency distribution in the spoligoforest.

Third, we detected host-pathogen associations in tuberculosis patients via genome-phenome data fusion using the Unified Biclustering Framework (UBF). We first restate host-pathogen association analysis as a biclustering problem, and then use the Unified Biclustering Framework to find statistically significant biclusters which represent pairs of spoligotype sets and country sets. We incorporate genetic distance between MTBC strains, spatial distance between TB patients, and time into domain knowledge, and factorize the joint datasets via coupled matrix-matrix and matrix-tensor factorization. We calculate the feature pattern similarity of spoligotype-country pairs and use this feature pattern similarity matrix as input to our novel density-invariant biclustering algorithm. Finally, we use average best-match score to find stable biclusters. The resulting biclusters verify some of the well-known associations between MTBC strains and geographic distribution of their hosts. Other biclusters suggest new associations to be investigated further by biologists.

Several aspects of these algorithmic data fusion methods can lead to new research problems. Next, we briefly describe two promising future directions in this research area.

6.1 Non-deterministic tensor decomposition

In the Tensor Clustering Framework (TCF) and Unified Biclustering Framework (UBF), we fit multiway models to tensors. Among these models, commonly used PARAFAC model is based on alternating least squares (ALS) method. However, ALS has drawbacks: It can fall into local minima, converge slowly, and can not recover the factor matrices accurately in the case of overfactoring. We aim to build a non-deterministic tensor decomposition algorithm to perform the same task with higher accuracy by escaping possible local minima to find global minima. We solve the permutation indeterminacy problem of PARAFAC model using factor match score defined by Acar et al. [79]. We also aim to solve the scaling indeterminacy problem by adding a Tikhonov regularization term to the original loss function of PARAFAC model.

In our initial experimental setup, we generated synthetic datasets in the form of tensors with varying size, rank, collinearity, homoscedastic noise level (i.e. constant variance, Gaussian noise), and heteroscedastic noise level (i.e. differing variance, Poisson noise). We also collected two real datasets: the fluorescence data analyzed by Riu and Bro [155], and multiple-biomarker tensor for *M. africanum* lineage which we used in sublineage structure analysis of MTBC in Chapter 3 [23]. Our initial algorithm to solve the non-deterministic tensor decomposition problem is Simulated Annealing with Adaptive Stepsize (SAAS). We start with initial factor matrices found by HOSVD, then we make the next move at a random direction. If the loss value decreases, we accept the next state, double the step size and move in the same direction in the next move. If the loss value increases, then we accept the next state with a probability depending on the temperature of the system, and step size remains constant. Otherwise, the new state is rejected and another random direction is picked. The step size is halved only when the temperature of the system is dropped. Our initial tests with this algorithm accurately decompose tensors when there is no homoscedastic or heteroscedastic noise, comparably as good as PARAFAC-ALS. When there is noise in the tensor, the performance of SAAS drastically drops and it can not recover the factor matrices, whereas PARAFAC-ALS can recover the factor matrices accurately, especially when there is no heteroscedastic noise. The shortcoming of our SAAS algorithm stems from the fact that it performs an exhaustive random search in continuous space, where there are infinitely many directions for the next move, and only a subset of all directions can lead to global minima. Moreover, SAAS algorithm is sensitive to noise. This suggests that we need to auto-tune the Tikhonov regularization constant based on the noise type and level of the tensor.

Fitting PARAFAC model is a nonlinear optimization problem which comes with several challenges. One factor affecting the speed of convergence of the tensor decomposition algorithms is the collinearity between the factors. Similarly, two factors which are collinear but have opposite signs can cancel out each other's contribution, which is also known as two-factor degeneracy [156]. This problem causes the loss function to decrease very slowly, while still not converging to global minimum. To speed up the algorithm, regularization and line search was suggested in earlier studies [157, 158]. Another particular case when PARAFAC-ALS fails to recover component matrices is the case of overfactoring. When more components than the rank of the tensor are used to decompose the tensor, PARAFAC-ALS fails most of the time [79]. Therefore, the need for a non-deterministic tensor decomposition method which solves the overfactoring problem constitutes another open research direction.

Extensions of non-deterministic tensor decomposition with various constraints also lead to new research directions. Many of the real-world data, including tuberculosis patients datasets used in this thesis, are nonnegative, and the corresponding component matrices have a physical meaning only when they are nonnegative [76]. Therefore, non-deterministic tensor decomposition with nonnegativity constraints on factor matrices are desirable. Similarly, multi-dimensional biological datasets are usually sparse. Therefore, sparse non-deterministic tensor decomposition methods are also beneficial and can lead to more accurate tensor factorizations [159]. Finally, datasets come with different forms of noise. Therefore, it is highly desirable to build a model selection framework for non-deterministic tensor decomposition in order to handle various noise types [76].

6.2 Host-pathogen association analysis

We restated host-pathogen association analysis as a biclustering problem earlier in Chapter 5 of this thesis. We can incorporate additional information from MTBC strains and TB patients in the joint dataset and factorize this dataset using the Unified Biclustering Framework. We can add other biomarkers of MTBC strains into domain knowledge such as MIRU and RFLP. We can also add new patient attributes such as risk factors including age group, homelessness, HIV status, MDR status, and ethnicity.

Transmission routes of tuberculosis follow the immigration map rather than the world map. Therefore, spatial proximity matrix can result in misleading results, e.g. US and Chinese patients can not be infected with the same strain based on the world map, but indeed they are infected due to immigration of Chinese population to the United States. On the other hand, an accurate immigration map reflecting the frequency of immigration routes between countries is hard to find, which is a data gathering problem. Therefore, if we base the spatial proximity matrix on the immigration map, we can accurately favor more likely transmission events.

Data factorization step in the Unified Biclustering Framework can be improved as well. ALS-based coupled matrix-matrix and matrix-tensor factorization algorithms we proposed in this thesis can be sped up using line search. This improvement on the speed will also improve the accuracy, since one of the convergence criterion of ALS is the maximum number of iterations, and line search increases the amount of movement at each step of ALS. This results in faster convergence to more accurate solutions.

We can also cluster genes based on their distribution to countries and number of patients they infected in each country. We can compare these clustering results to obtained biclustering results. We can consider the clustering result as the ground truth for classification of spoligotypes, and use it as part of an external measure such as F-measure in bicluster validation step. This will base statistically significant bicluster selection on an external measure instead of an internal measure, which can lead to improved bicluster selection. In the presence of limited ground truth for biclusters, we can also use semi-supervised biclustering to find new host-pathogen associations [160]. That will improve the quality of identified biclusters and lead to more coherent patient-strain relationships.

REFERENCES

[1] World Health Organization (WHO) Report, “Global tuberculosis control : epidemiology, strategy, financing,” 2009.

[2] World Health Organization (WHO), “Tuberculosis: Fact Sheet No 104,” November 2010.

[3] M. P. Golden and H. R. Vikram, “Extrapulmonary Tuberculosis: An Overview,” *Amer. Family Physician*, vol. 72, no. 9, pp. 1761–1768, 2005.

[4] S. Gagneux, K. DeRiemer, *et al.*, “Variable host-pathogen compatibility in *Mycobacterium tuberculosis*,” *Proc. Nat. Academy of Sciences (PNAS)*, vol. 103, no. 8, pp. 2869–2873, 2006.

[5] CDC, “Reported Tuberculosis in the United States, 2008,” September 2009.

[6] C. Colijn, T. Cohen, *et al.*, “Mathematical Models of Tuberculosis: accomplishments and future challenges,” in *Proc. BIOMAT Int. Symp. on Math. and Computational Biology*, pp. 123–148, World Scientific Publisher, 2006.

[7] J. P. Aparicio, A. F. Capurro, *et al.*, “Transmission and Dynamics of Tuberculosis on Generalized Households,” *J. Theoretical Biology*, vol. 206, no. 3, pp. 327 – 341, 2000.

[8] G. L. Snider, “Tuberculosis then and now: A personal perspective on the last 50 years,” *Ann. Internal Medicine*, vol. 126, no. 3, pp. 237–243, 1997.

[9] C. Ozcaglar *et al.*, “Epidemiological models of *Mycobacterium tuberculosis* complex infections,” *Math. Biosciences*, vol. 236, pp. 77 – 96, April 2012.

[10] I. Vitol, *Mathematical models for Mycobacterium tuberculosis complex genotyping and patient data.* Ph.D. dissertation, Comp. Sci., Rensselaer Polytechnic Inst., Troy, NY, 2006.

[11] S. T. Cole, R. Brosch, *et al.*, “Deciphering the biology of *Mycobacterium tuberculosis* from the complete genome sequence,” *Nature*, vol. 393, pp. 537–544, June 1998.

[12] T. Garnier, K. Eiglmeier, *et al.*, “The complete genome sequence of *Mycobacterium bovis*,” *Proc. Nat. Academy of Sciences (PNAS)*, vol. 100, no. 13, pp. 7877–7882, 2003.

[13] B. Mathema, N. Kurepina, *et al.*, “Molecular epidemiology of tuberculosis: Current insights,” *Clin. Microbiol. Rev.*, vol. 19, no. 4, pp. 658–685, 2006.

[14] A. Shabbeer, C. Ozcaglar, *et al.*, “Web tools for molecular epidemiology of tuberculosis,” *Infection, Genetics and Evolution*, vol. 12, no. 4, pp. 767 – 781, 2012.

[15] P. F. Barnes and M. D. Cave, “Molecular epidemiology of tuberculosis,” *New England J. Medicine*, vol. 349, no. 12, pp. 1149–1156, 2003.

[16] E. Mazars, S. Lesjean, *et al.*, “High-resolution minisatellite-based typing as a portable approach to global analysis of *Mycobacterium tuberculosis* molecular epidemiology,” *Proc. Nat. Academy of Sciences (PNAS)*, vol. 98, no. 4, pp. 1901–1906, 2001.

[17] J. Kamerbeek, L. Schouls, *et al.*, “Simultaneous detection and strain differentiation of *Mycobacterium tuberculosis* for diagnosis and epidemiology,” *J. Clin. Microbiology*, vol. 35, no. 4, pp. 907–914, 1997.

[18] P. Supply, C. Allix, *et al.*, “Proposal for Standardization of Optimized Mycobacterial Interspersed Repetitive Unit-Variable-Number Tandem Repeat Typing of *Mycobacterium tuberculosis*,” *J. Clin. Microbiology*, vol. 44, no. 12, pp. 4498–4510, 2006.

[19] I. Filliol, J. Driscoll, *et al.*, “Global distribution of *Mycobacterium tuberculosis* spoligotypes,” *Emerging Infectious Diseases*, vol. 8, no. 11, pp. 1347–1350, 2002.

[20] M. Aminian, A. Shabbeer, *et al.*, “A conformal Bayesian network for classification of *Mycobacterium tuberculosis* complex lineages,” *BMC Bioinformatics*, vol. 11, no. Suppl 3, p. S4, 2010.

[21] M. Aminian, A. Shabbeer, *et al.*, “Knowledge-based Bayesian network for the classification of *Mycobacterium tuberculosis* complex sublineages,” in *Proc. 2nd ACM Conf. Bioinformatics, Computational Biology and Biomedicine*, pp. 201–208, 2011.

[22] M. Aminian, A. Shabbeer, *et al.*, “Incorporating biology rules of thumb into Bayesian networks.” *J. Bioinformatics and Computational Biology*, in press, 2012.

[23] C. Ozcaglar *et al.*, “Sublineage structure analysis of *Mycobacterium tuberculosis* complex strains using multiple-biomarker tensors,” *BMC Genomics*, vol. 12, no. Suppl 2, p. S1, 2011.

[24] C. Ozcaglar *et al.*, “Examining the sublineage structure of *Mycobacterium tuberculosis* complex strains with multiple-biomarker tensors,” in *Proc. 2010*

IEEE Int. Conf. Bioinformatics and Biomedicine (BIBM), (Hong Kong), pp. 543–548, 2010.

[25] C. Ozcaglar *et al.*, "A clustering framework for *Mycobacterium tuberculosis* complex strains using multiple-biomarker tensors," tech. rep., Dept. of Comp. Sci., Rensselaer Polytechnic Inst., Troy, NY, Tech. Rep. 10-08, 2010.

[26] A. Shabbeer, L. S. Cowan, *et al.*, "TB-Lineage: An online tool for classification and analysis of strains of *Mycobacterium tuberculosis* complex," *Infection, Genetics and Evolution*, vol. 12, no. 4, pp. 789 – 797, 2012.

[27] K. Brudey, J. Driscoll, *et al.*, "*Mycobacterium tuberculosis* complex genetic diversity: mining the fourth international spoligotyping database (SpolDB4) for classification, population genetics and epidemiology," *BMC Microbiology*, vol. 6, no. 1, p. 23, 2006.

[28] E. Legrand, I. Filliol, *et al.*, "Use of spoligotyping to study the evolution of the Direct Repeat locus by IS6110 transposition in *Mycobacterium tuberculosis*," *J. Clin. Microbiology*, vol. 39, no. 4, pp. 1595–1599, 2001.

[29] Z. Fang, N. Morrison, *et al.*, "IS6110 transposition and evolutionary scenario of the Direct Repeat locus in a group of closely related *Mycobacterium tuberculosis* strains," *J. Bacteriology*, vol. 180, no. 8, pp. 2102–2109, 1998.

[30] L. Fenner, B. Malla, *et al.*, "“Pseudo-Beijing”: Evidence for convergent evolution in the Direct Repeat region of *Mycobacterium tuberculosis*," *PLoS ONE*, vol. 6, p. e24737, Sep. 2011.

[31] A. E. Hirsh, A. G. Tsolaki, *et al.*, "Stable association between strains of *Mycobacterium tuberculosis* and their human host populations," *Proc. Nat. Academy of Sciences (PNAS)*, vol. 101, no. 14, pp. 4871–4876, 2004.

[32] C. Ozcaglar *et al.*, "Data-driven insights into deletions of *Mycobacterium tuberculosis* complex chromosomal DR Region using spoligoforests," in *Proc. 2011 IEEE Int. Conf. Bioinformatics and Biomedicine (BIBM)*, (Atlanta, GA), pp. 75–82, 2011.

[33] C. Ozcaglar *et al.*, "Inferred spoligoforest topology unravels spatially bimodal distribution of mutations in the DR region." *IEEE Trans. NanoBioscience*, in press, 2012.

[34] C. Ozcaglar, B. Yener, *et al.*, "Host-pathogen association analysis of tuberculosis patients via Unified Biclustering Framework," tech. rep., Dept. Comp. Sci., Rensselaer Polytechnic Inst., Troy, NY, Tech. Rep. 12-05, 2012.

[35] J. C. Venter, M. D. Adams, *et al.*, "The Sequence of the Human Genome," *Science*, vol. 291, no. 5507, pp. 1304–1351, 2001.

[36] J. Tang, C. Tan, *et al.*, “Integrating post-genomic approaches as a strategy to advance our understanding of health and disease,” *Genome Medicine*, vol. 1, no. 3, p. 35, 2009.

[37] J. Han and M. Kamber, *Data Mining: Concepts and Techniques.* San Francisco, CA: Morgan Kaufmann, 2006.

[38] P.-N. Tan, M. Steinbach, *et al.*, *Introduction to Data Mining.* Boston, MA: Addison-Wesley Longman Publishing Co., Inc., 2005.

[39] M. P. S. Brown, W. N. Grundy, *et al.*, “Knowledge-based analysis of microarray gene expression data by using support vector machines,” *Proc. Nat. Academy of Sciences (PNAS)*, vol. 97, no. 1, pp. 262–267, 2000.

[40] D. V. Nguyen and D. M. Rocke, “Multi-class cancer classification via partial least squares with gene expression profiles,” *Bioinformatics*, vol. 18, no. 9, pp. 1216–1226, 2002.

[41] J. Quackenbush, “Computational analysis of microarray data,” *Nature Reviews Genetics*, vol. 2, no. 6, pp. 418–427, 2001.

[42] M. B. Eisen, P. T. Spellman, *et al.*, “Cluster analysis and display of genome-wide expression patterns,” *Proc. Nat. Academy of Sciences (PNAS)*, vol. 95, no. 25, pp. 14863–14868, 1998.

[43] S. Ferdinand, G. Valétudie, *et al.*, “Data mining of *Mycobacterium tuberculosis* complex genotyping results using mycobacterial interspersed repetitive units validates the clonal structure of spoligotyping-defined families,” *Research in Microbiology*, vol. 155, no. 8, pp. 647–654, 2004.

[44] I. Vitol, J. Driscoll, *et al.*, “Identifying *Mycobacterium tuberculosis* complex strain families using spoligotypes,” *Infection, Genetics and Evolution*, vol. 6, no. 6, pp. 491–504, 2006.

[45] C. Borile, M. Labarre, *et al.*, “Using affinity propagation for identifying subspecies among clonal organisms: lessons from *M. tuberculosis*,” *BMC Bioinformatics*, vol. 12, no. 1, p. 224, 2011.

[46] B. Mirkin, *Mathematical classification and clustering.* Dordrecht, Netherlands: Kluwer Academic Press, 1996.

[47] J. A. Hartigan, “Direct Clustering of a Data Matrix,” *J. Amer. Statistical Assoc.*, vol. 67, no. 337, pp. 123–129, 1972.

[48] S. C. Madeira and A. L. Oliveira, “Biclustering algorithms for biological data analysis: A survey,” *IEEE/ACM Trans. Computational Biology and Bioinformatics*, vol. 1, pp. 24–45, Jan. 2004.

[49] Y. Cheng and G. M. Church, "Biclustering of expression data," in *Proc. Intelligent Syst. for Molecular Biology (ISMB)*, pp. 93–103, 2000.

[50] J. Yang, H. Wang, *et al.*, "Enhanced biclustering on expression data," in *Proc. 3rd IEEE Symp. Bioinformatics and Bioengineering (BIBE'03)*, pp. 321–327, 2003.

[51] L. Lazzeroni and A. Owen, "Plaid models for gene expression data," *Statistica Sinica*, vol. 12, pp. 61–86, 2000.

[52] G. Getz, E. Levine, *et al.*, "Coupled two-way clustering analysis of gene microarray data," *Proc. Nat. Academy of Sciences (PNAS)*, vol. 97, no. 22, pp. 12079–12084, 2000.

[53] C. Tang, L. Zhang, *et al.*, "Interrelated two-way clustering: An unsupervised approach for gene expression data analysis," in *Proc. 2nd IEEE Int. Symp. Bioinformatics and Bioengineering (BIBE)*, pp. 41–48, 2001.

[54] I. S. Dhillon, "Co-clustering documents and words using bipartite spectral graph partitioning," in *Proc. 7th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD '01)*, pp. 269–274, 2001.

[55] Y. Kluger, R. Basri, *et al.*, "Spectral biclustering of microarray data: coclustering genes and conditions," *Genome Research*, vol. 13, no. 4, pp. 703–716, 2003.

[56] Q. Sheng, Y. Moreau, *et al.*, "Biclustering microarray data by Gibbs sampling," *Bioinformatics*, vol. 19, no. suppl 2, pp. ii196–ii205, 2003.

[57] A. Ben-Dor, B. Chor, *et al.*, "Discovering local structure in gene expression data: the order-preserving submatrix problem," in *Proc. 6th Ann. Int. Conf. Computational Biology (RECOMB '02)*, pp. 49–57, 2002.

[58] A. Tanay, R. Sharan, *et al.*, "Discovering statistically significant biclusters in gene expression data," *Bioinformatics*, vol. 18, no. suppl 1, pp. S136–S144, 2002.

[59] T. M. Murali and S. Kasif, "Extracting conserved gene expression motifs from gene expression data," in *Proc. Pac. Symp. Biocomputing*, pp. 77–88, 2003.

[60] A. Prelic, S. Bleuler, *et al.*, "A systematic comparison and evaluation of biclustering methods for gene expression data," *Bioinformatics*, vol. 22, no. 9, pp. 1122–1129, 2006.

[61] P. Dao, R. Colak, *et al.*, "Inferring cancer subnetwork markers using density-constrained biclustering," *Bioinformatics*, vol. 26, no. 18, pp. i625–i631, 2010.

[62] P. Baldi and G. W. Hatfield, *DNA Microarrays and Gene Regulation.* Cambridge, United Kingdom: Cambridge University Press, 2001.

[63] A. Tanay, R. Sharan, *et al.*, “Revealing modularity and organization in the yeast molecular network by integrated analysis of highly heterogeneous genomewide data,” *Proc. Nat. Academy of Sciences (PNAS)*, vol. 101, no. 9, pp. 2981–2986, 2004.

[64] R. Colak, F. Moser, *et al.*, “Module discovery by exhaustive search for densely connected, co-expressed regions in biomolecular interaction networks,” *PLoS ONE*, vol. 5, p. e13348, Oct. 2010.

[65] E. Acar and B. Yener, “Unsupervised multiway data analysis: A literature survey,” *IEEE Trans. Knowledge and Data Eng.*, vol. 21, no. 1, pp. 6–20, 2009.

[66] E. Acar, *Understanding epilepsy seizure structure using tensor analysis.* Ph.D. dissertation, Dept. Comp. Sci., Rensselaer Polytechnic Inst., Troy, NY, 2008.

[67] H. A. L. Kiers, “Towards a standardized notation and terminology in multiway analysis,” *J. Chemometrics*, vol. 14, no. 3, pp. 105–122, 2000.

[68] J. Carroll and J.-J. Chang, “Analysis of individual differences in multidimensional scaling via an n-way generalization of “Eckart-Young” decomposition,” *Psychometrika*, vol. 35, pp. 283–319, 1970.

[69] R. A. Harshman, “Foundations of the PARAFAC procedure: Models and conditions for an “explanatory” multi-modal factor analysis,” *UCLA Working Papers in Phonetics*, vol. 16, no. 1, p. 84, 1970.

[70] Joseph B. Kruskal, “Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics,” *Linear Algebra and its Applications*, vol. 18, no. 2, pp. 95 – 138, 1977.

[71] L. R. Tucker, “Implications of factor analysis of three-way matrices for measurement of change,” in *Problems in Measuring Change*, pp. 122–137, Madison WI: University of Wisconsin Press, 1963.

[72] L. R. Tucker, “Some mathematical notes on three-mode factor analysis,” *Psychometrika*, vol. 31, pp. 279–311, 1966.

[73] P. M. Kroonenberg, *Applied Multiway Data Analysis.* Hoboken, NJ: Wiley, 2008.

[74] T. G. Kolda and B. W. Bader, “Tensor decompositions and applications,” *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.

[75] T. G. Kolda, “Multilinear operators for higher-order decompositions,” tech. rep., Sandia National Laboratories, Albuquerque, NM and Livermore, CA, Tech. Rep. SAND2006-2081, April 2006.

[76] A. Cichocki, R. Zdunek, *et al.*, *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation.* Chichester, United Kingdom: Wiley, 2009.

[77] A. K. Smilde, R. Bro, *et al.*, *Multi-way Analysis: Applications in the Chemical Sciences.* Chichester, United Kingdom: Wiley, 2004.

[78] G. Tomasi and R. Bro, “A comparison of algorithms for fitting the PARAFAC model,” *Computational Stat. & Data Anal.*, vol. 50, no. 7, pp. 1700–1734, 2006.

[79] E. Acar, D. M. Dunlavy, *et al.*, “A scalable optimization approach for fitting canonical tensor decompositions,” *J. Chemometrics*, vol. 25, no. 2, pp. 67–86, 2011.

[80] L. D. Lathauwer, B. D. Moor, *et al.*, “On the best rank-1 and rank-(r_1,r_2,...,r_n) approximation of higher-order tensors,” *SIAM J. Matrix Anal. Appl.*, vol. 21, pp. 1324–1342, Mar. 2000.

[81] H. Wang and N. Ahuja, “A tensor approximation approach to dimensionality reduction,” *Int. J. Comput. Vision*, vol. 76, pp. 217–229, Mar. 2008.

[82] P. D. Turney, “Empirical evaluation of four tensor decomposition algorithms,” tech. rep., Inst. Information Technology, National Research Council of Canada, NRC Tech. Report ERB-1152, November 2007.

[83] M. Mørup, “Applications of tensor (multiway array) factorizations and decompositions in data mining,” *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 1, no. 1, pp. 24–40, 2011.

[84] E. Acar, C. Aykut-Bingol, *et al.*, “Multiway analysis of epilepsy tensors,” *Bioinformatics*, vol. 23, no. 13, pp. i10–i18, 2007.

[85] L. Omberg, G. H. Golub, *et al.*, “A tensor higher-order singular value decomposition for integrative analysis of DNA microarray data from different studies,” *Proc. Nat. Academy of Sciences (PNAS)*, vol. 104, no. 47, pp. 18371–18376, 2007.

[86] C. Muralidhara, A. M. Gross, *et al.*, “Tensor decomposition reveals concurrent evolutionary convergences and divergences and correlations with structural motifs in ribosomal RNA,” *PLoS ONE*, vol. 6, no. 4, p. e18768, 2011.

[87] B. Yener, E. Acar, *et al.*, "Multiway modeling and analysis in stem cell systems biology," *BMC Syst. Biology*, vol. 2, no. 1, p. 63, 2008.

[88] M. J. Zvelebil and J. O. Baum, *Understanding Bioinformatics.* New York, NY: Garland Science, 2008.

[89] J. Reyes, A. Francis, *et al.*, "Models of deletion for visualizing bacterial variation: an application to tuberculosis spoligotypes," *BMC Bioinformatics*, vol. 9, no. 1, p. 496, 2008.

[90] C. D. Michener and R. R. Sokal, "A quantitative approach to a problem in classification," *Evolution*, vol. 11, no. 2, pp. 130–162, 1957.

[91] N. Saitou and M. Nei, "The neighbour-joining method: a new method for reconstructing phylogenetic trees," *Molecular Biology and Evolution*, vol. 4, no. 4, pp. 406–425, 1987.

[92] W. M. Fitch and E. Margoliash, "Construction of phylogenetic trees," *Science*, vol. 155, no. 3760, pp. 279–284, 1967.

[93] J. Felsenstein, *Inferring Phylogenies.* Sunderland, MA: Sinauer Associates, 2003.

[94] A. W. F. Edwards and L. L. Cavalli-Sforza, "The reconstruction of evolution," *Ann. Human Genetics*, vol. 18, no. 2, pp. 104–105, 1963.

[95] A. W. F. Edwards and L. L. Cavalli-Sforza, "Reconstruction of evolutionary trees," in *Phenetic and Phylogenetic Classification*, pp. 67–76, London, United Kingdom: Systematics Association, 1964.

[96] J. P. Huelsenbeck, F. Ronquist, *et al.*, "Bayesian inference of phylogeny and its impact on evolutionary biology," *Science*, vol. 294, no. 5550, pp. 2310–2314, 2001.

[97] S. Gagneux and P. M. Small, "Global phylogeography of *Mycobacterium tuberculosis* and implications for tuberculosis product development," *The Lancet Infectious Diseases*, vol. 7, no. 5, pp. 328 – 337, 2007.

[98] B. Asiimwe, *Molecular characterization of Mycobacterium tuberculosis complex in Kampala, Uganda.* Ph.D. dissertation, Dept. Microbiology, Tumor and Cell Biology, Makerere Univ., Kampala, Uganda, 2008.

[99] O. Rubel, G. H. Weber, *et al.*, "Integrating Data Clustering and Visualization for the Analysis of 3D Gene Expression Data," *IEEE/ACM Trans. Computational Biology and Bioinformatics*, vol. 7, no. 1, pp. 64–79, 2010.

[100] J. Handl, J. Knowles, *et al.*, "Computational cluster validation in post-genomic data analysis," *Bioinformatics*, vol. 21, no. 15, pp. 3201–3212, 2005.

[101] R. Giancarlo, D. Scaturro, *et al.*, "Computational cluster validation for microarray data analysis: experimental assessment of Clest, Consensus Clustering, Figure of Merit, Gap Statistics and Model Explorer," *BMC Bioinformatics*, vol. 9, no. 1, p. 462, 2008.

[102] H.-P. Kriegel, P. Kröger, *et al.*, "Clustering High-Dimensional Data: A Survey on Subspace Clustering, Pattern-Based Clustering, and Correlation Clustering," *ACM Trans. Knowl. Discov. Data*, vol. 3, no. 1, pp. 1–58, 2009.

[103] O. Alter and G. H. Golub, "Reconstructing the pathways of a cellular system from genome-scale signals by using matrix and tensor computations," *Proc. Nat. Academy of Sciences (PNAS)*, vol. 102, no. 49, pp. 17559–17564, 2005.

[104] R. M. Warren, E. M. Streicher, *et al.*, "Microevolution of the Direct Repeat region of *Mycobacterium tuberculosis*: Implications for interpretation of spoligotyping data," *J. Clin. Microbiology*, vol. 40, no. 12, pp. 4457–4465, 2002.

[105] C. A. Andersson and R. Bro, "The N-way toolbox for MATLAB," *Chemometrics and Intelligent Laboratory Systems*, vol. 52, no. 1, pp. 1 – 4, 2000.

[106] Eigenvector Research, Inc., "PLS Toolbox." http://www.eigenvector.com/. Last Accessed: March 2011.

[107] P. M. Kroonenberg, "Three mode component models: A survey of the literature," *Statistica Applicata*, vol. 4, no. 4, pp. 619–633, 1992.

[108] R. Bro and H. Kiers, "A new efficient method for determining the number of components in PARAFAC models," *J. Chemometrics*, vol. 17, no. 5, pp. 274–286, 2003.

[109] D. Arthur and S. Vassilvitskii, "K-means++: The advantages of careful seeding," in *Proc. Eighteenth Annu. ACM-SIAM Symp. Discrete Algorithms (SODA'07)*, pp. 1027–1035, 2007.

[110] M. Halkidi, Y. Batistakis, *et al.*, "Cluster validity methods: part I," *SIGMOD Rec.*, vol. 31, no. 2, pp. 40–45, 2002.

[111] S. Ben-David, U. V. Luxburg, *et al.*, "A sober look at clustering stability," in *Proc. Conf. Learning Theory (COLT)*, (Pittsburgh, PA), pp. 5–19, 2006.

[112] J. Hopcroft, O. Khan, *et al.*, “Natural communities in large linked networks,” in *Proc. 9th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD ’03)*, (Washington, DC), pp. 541–546, 2003.

[113] R. Tibshirani, G. Walther, *et al.*, “Estimating the number of clusters in a dataset via the gap statistic,” *J. Roy. Statistical Soc.*, vol. 63, no. 2, pp. 411–423, 2000.

[114] S. Dudoit and J. Fridlyand, “A prediction-based resampling method for estimating the number of clusters in a dataset,” *Genome Biology*, vol. 3, no. 7, pp. 1–21, 2002.

[115] M. Yan and K. Ye, “Determining the number of clusters using the weighted gap statistic,” *Biometrics*, vol. 63, no. 4, pp. 1031–7, 2007.

[116] R. Bro, “Multiway calibration. Multilinear PLS,” *J. Chemometrics*, vol. 10, no. 1, pp. 47–61, 1996.

[117] A. K. Smilde, “Comments on multilinear PLS,” *J. Chemometrics*, vol. 11, no. 5, pp. 367–377, 1997.

[118] S. de Jong, “Regression coefficients in multilinear PLS,” *J. Chemometrics*, vol. 12, no. 1, pp. 77–81, 1998.

[119] R. Bro, A. K. Smilde, *et al.*, “On the difference between low-rank and subspace approximation: improved model for multi-linear PLS regression,” *Chemometrics and Intelligent Laboratory Syst.*, vol. 58, no. 1, pp. 3 – 13, 2001.

[120] R. Bro and A. K. Smilde, “Centering and scaling in component analysis,” *J. Chemometrics*, vol. 17, no. 1, pp. 16–33, 2003.

[121] A. L. Gibson, R. C. Huard, *et al.*, “Application of sensitive and specific molecular methods to uncover global dissemination of the major RD^{Rio} sublineage of the Latin American-Mediterranean *Mycobacterium tuberculosis* spoligotype family,” *J. Clin. Microbiology*, vol. 46, no. 4, pp. 1259–1267, 2008.

[122] A. Bertoni and G. Valentini, “Model order selection for bio-molecular data clustering,” *BMC Bioinformatics*, vol. 8, no. Suppl 2, p. S7, 2007.

[123] R. Brosch, S. V. Gordon, *et al.*, “A new evolutionary scenario for the *Mycobacterium tuberculosis* complex,” *Proc. Nat. Academy of Sciences (PNAS)*, vol. 99, no. 6, pp. 3684–3689, 2002.

[124] M. M. Tanaka and A. R. Francis, “Methods of quantifying and visualizing outbreaks of tuberculosis using genotypic information,” *Infection, Genetics and Evolution*, vol. 5, no. 1, pp. 35–43, 2005.

[125] A. Grant, C. Arnold, *et al.*, "Mathematical modelling of *Mycobacterium tuberculosis* VNTR loci estimates a very slow mutation rate for the repeats," *J. Molecular Evolution*, vol. 66, no. 6, pp. 565–574, 2008.

[126] E. R. Gansner and S. C. North, "An open graph visualization system and its applications to software engineering," *Software-Practice & Experience*, vol. 30, no. 11, pp. 1203–1233, 2000.

[127] J. F. Reyes and M. M. Tanaka, "Mutation rates of spoligotypes and variable numbers of tandem repeat loci in *Mycobacterium tuberculosis*," *Infection, Genetics and Evolution*, vol. 10, no. 7, pp. 1046 – 1051, 2010.

[128] J. D. A. van Embden, T. van Gorkom, *et al.*, "Genetic variation and evolutionary origin of the direct repeat locus of *Mycobacterium tuberculosis* complex bacteria," *J. Bacteriol.*, vol. 182, no. 9, pp. 2393–2401, 2000.

[129] J. Zhang, E. Abadia, *et al.*, "*Mycobacterium tuberculosis* complex CRISPR genotyping: improving efficiency, throughput and discriminative power of 'spoligotyping' with new spacers and a microbead-based hybridization assay," *J. Medical Microbiology*, vol. 59, no. 3, pp. 285–294, 2010.

[130] P. Supply, J. Magdalena, *et al.*, "Identification of novel intergenic repetitive units in a mycobacterial two-component system operon," *Molecular Microbiology*, vol. 26, no. 5, pp. 991–1003, 1997.

[131] P. Supply, E. Mazars, *et al.*, "Variable human minisatellite-like regions in the *Mycobacterium tuberculosis* genome," *Molecular Microbiology*, vol. 36, no. 3, pp. 762–771, 2000.

[132] J. Camin and R. Sokal, "A method for deducting branching sequences in phylogeny," *Evolution*, vol. 19, no. 3, pp. 311–326, 1965.

[133] M. Kimura and T. Ohta, "Stepwise mutation model and distribution of allelic frequencies in a finite population.," *Proc. Nat. Academy of Sciences (PNAS)*, vol. 75, no. 6, pp. 2868–2872, 1978.

[134] T. Wirth, F. Hildebrand, *et al.*, "Origin, spread and demography of the *Mycobacterium tuberculosis* complex," *PLoS Pathogen*, vol. 4, no. 9, p. e1000160, 2008.

[135] G. Pavlopoulos, M. Secrier, *et al.*, "Using graph theory to analyze biological networks," *BioData Mining*, vol. 4, no. 1, pp. 10+, 2011.

[136] G. Lima-Mendez and J. Helden, "The powerful law of the power law and other myths in network biology," *Molecular BioSystems*, vol. 5, no. 12, pp. 1482–1493, 2009.

[137] A. Clauset, C. R. Shalizi, *et al.*, "Power-law distributions in empirical data," *SIAM Review*, vol. 51, pp. 661+, February 2009.

[138] C. Tang, J. Reyes, *et al.*, "spolTools: Online utilities for analyzing spoligotypes of the *Mycobacterium tuberculosis* complex," *Bioinformatics*, vol. 24, no. 20, pp. 2414–2415, 2008.

[139] S. Sreevatsan, X. Pan, *et al.*, "Restricted structural gene polymorphism in the *Mycobacterium tuberculosis* complex indicates evolutionarily recent global dissemination," *Proc. Nat. Academy of Sciences (PNAS)*, vol. 94, no. 18, pp. 9869–9874, 1997.

[140] C. Demay, B. Liens, *et al.*, "SITVITWEB - A publicly available international multimarker database for studying *Mycobacterium tuberculosis* genetic diversity and molecular epidemiology," *Infection, Genetics and Evolution*, vol. 12, no. 4, pp. 755 – 766, 2012.

[141] A. van der Zanden, K. Kremer, *et al.*, "Improvement of differentiation and interpretability of spoligotyping for *Mycobacterium tuberculosis* complex isolates by introduction of new spacer oligonucleotides," *J. Clin. Microbiology*, vol. 40, no. 12, pp. 4628–4639, 2002.

[142] B. Mathema, N. Kurepina, *et al.*, "Epidemiologic consequences of microvariation in *Mycobacterium tuberculosis*," *J. Infectious Diseases*, vol. 205, no. 6, pp. 964–974, 2012.

[143] K. Bennett, C. Ozcaglar, *et al.*, "Visualization of tuberculosis patient and *Mycobacterium tuberculosis* complex genotype data via host-pathogen maps," in *Proc. 2011 IEEE Int. Conf. Bioinformatics and Biomedicine Workshops (BIBMW)*, pp. 124–129, nov. 2011.

[144] S. Yu, T. Falck, *et al.*, "L2-norm multiple kernel learning and its application to biomedical data fusion," *BMC Bioinformatics*, vol. 11, no. 1, p. 309, 2010.

[145] G. R. G. Lanckriet, T. De Bie, *et al.*, "A statistical framework for genomic data fusion," *Bioinformatics*, vol. 20, no. 16, pp. 2626–2635, 2004.

[146] G. R. G. Lanckriet, N. Cristianini, *et al.*, "Kernel-based integration of genomic data using semidefinite programming," in *Kernel Methods in Computational Biology*, pp. 231–263, Cambridge, MA: MIT Press, 2004.

[147] S. Aerts, D. Lambrechts, *et al.*, "Gene prioritization through genomic data fusion," *Nature Biotechnology*, vol. 24, no. 5, pp. 537–544, 2006.

[148] K. Lage, E. O. Karlberg, *et al.*, "A human phenome-interactome network of protein complexes implicated in genetic disorders," *Nature Biotechnology*, vol. 25, no. 3, pp. 309–316, 2007.

[149] T. H. Cormen, C. E. Leiserson, *et al.*, *Introduction to Algorithms.* Cambridge, MA: The MIT Press, 2001.

[150] E. Acar, T. G. Kolda, *et al.*, "All-at-once optimization for coupled matrix and tensor factorizations," *ArXiv e-prints*, May 2011.

[151] X. Liu and L. Wang, "Computing the maximum similarity biclusters of gene expression data," *Bioinformatics*, vol. 23, no. 1, pp. 50–56, 2007.

[152] P. J. Bifani, B. Mathema, *et al.*, "Global dissemination of the *Mycobacterium tuberculosis* W-Beijing family strains," *Trends in Microbiology*, vol. 10, no. 1, pp. 45 – 52, 2002.

[153] J. R. Glynn, J. Whiteley, *et al.*, "Worldwide occurrence of Beijing/W strains of *Mycobacterium tuberculosis*: a systematic review," *Emerging Infectious Diseases*, vol. 8, no. 8, pp. 843 – 849, 2002.

[154] M. Zervakis, M. Blazadonakis, *et al.*, "Outcome prediction based on microarray analysis: a critical perspective on methods," *BMC Bioinformatics*, vol. 10, no. 1, p. 53, 2009.

[155] J. Riu and R. Bro, "Jack-knife technique for outlier detection and estimation of standard errors in PARAFAC models," *Chemometrics and Intelligent Laboratory Syst.*, vol. 65, no. 1, pp. 35–49, 2002.

[156] B. C. Mitchell and D. S. Burdick, "Slowly converging PARAFAC sequences: Swamps and two-factor degeneracies," *J. Chemometrics*, vol. 8, no. 2, pp. 155–168, 1994.

[157] P. Paatero, "A weighted non-negative least squares algorithm for three-way "PARAFAC" factor analysis," *Chemometrics and Intelligent Laboratory Syst.*, vol. 38, no. 2, pp. 223–242, 1997.

[158] R. Bro, *Multi-way Analysis in the Food Industry - Models, Algorithms, and Applications.* Ph.D. dissertation, Dept. Dairy and Food Science, Royal Veterinary and Agricultural Univ., Frederiksberg, Denmark, 1998.

[159] M. Mørup, L. K. Hansen, *et al.*, "Algorithms for sparse nonnegative Tucker decompositions," *Neural Computation*, vol. 20, no. 8, pp. 2112–2131, 2008.

[160] L. Teng and K. Tan, "Finding combinatorial histone code by semi-supervised biclustering," *BMC Genomics*, vol. 13, no. 1, p. 301, 2012.