A clustering framework for *Mycobacterium tuberculosis* complex strains using multiple-biomarker tensors

Cagri Ozcaglar[1], Amina Shabbeer[1], Scott Vandenberg[3], Bülent Yener[1], Kristin P. Bennett[1,2]
(1) Computer Science Department and (2) Mathematical Sciences Department, Rensselaer Polytechnic Institute
(3) Computer Science Department, Siena College
ozcagc2@cs.rpi.edu, shabba@cs.rpi.edu, vandenberg@siena.edu, yener@cs.rpi.edu, bennek@rpi.edu

Abstract—Strains of the *Mycobacterium tuberculosis* complex (MTBC) can be classified into coherent lineages of similar traits based on their genotype. We present a tensor clustering framework to group MTBC strains into sublineages of the known major lineages based on two biomarkers: spacer oligonucleotide type (spoligotype) and mycobacterial interspersed repetitive units (MIRU). We represent genotype information of MTBC strains in a high-dimensional array in order to include information about spoligotype, MIRU, and their coexistence using multiple-biomarker tensors. We use multiway models to transform this multidimensional data about the MTBC strains into two-dimensional arrays and use the resulting score vectors in a stable partitive clustering algorithm to classify MTBC strains into sublineages within their major lineage. We validate clusterings using cluster stability and accuracy measures, and find stabilities of each cluster. Based on validated clustering results, we present a sublineage structure of MTBC strains and compare it to the sublineage structures of SpolDB4 and MIRU-VNTR*plus*.

Index Terms—Tuberculosis, *Mycobacterium tuberculosis* complex, multiway models, clustering, cluster validation

This technical report is the online supplement of the paper titled "Examining the sublineage structure of Mycobacterium tuberculosis *complex strains with multiple-biomarker tensors" published in the IEEE International Conference on Bioinformatics & Biomedicine 2010.*

1 Introduction

Tuberculosis (TB) is a bacterial disease caused by *Mycobacterium tuberculosis* complex (MTBC), which 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-VNTRs), IS6110 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 and control and examining host-pathogen relationships [1]. The major lineages of MTBC are *M. africanum*, *M. canettii*, *M. microti*, *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). These major lineages can be definitively characterized using LSPs [2], but typically only MIRU and spoligotypes are collected for the purpose 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 [3]–[5].

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 vast majority of them labeled and classified into 62 sublineages [6], but many of these are considered to be "potentially phylogeographically-specific MTBC genotype families". 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* [7]. Now large databases using both MIRU and spoligotypes 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 MIRU and spoligotypes are critical for both TB control and research.

This study uses unsupervised multiway analysis to examine the sublineage structure of MTBC on the basis of spoligotype and MIRU patterns. The proposed method reveals structure not captured in SpolDB4 spoligotype families. When MIRU patterns are considered, SpolDB4 families that may be well supported by spoligotype signatures, become ambiguous, or may allow further subdivision. A key issue is how to combine spoligotype and MIRU into a single unsupervised learning model. A spoligotype-only tool, SPOTCLUST, was used to find MTBC sublineages using an unsupervised probabilistic model reflecting spoligotype evolution [8]. 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 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, possibly yielding different results.

In this study, we develop a tensor clustering framework for sublineage classification of MTBC strains labeled by

major lineages. We generate multiple-biomarker tensors of MTBC strains and apply multiway models for dimensionality reduction. The model accurately captures spoligotype evolutionary dynamics by using contiguous deletions of spacers. The tensor transforms spoligotypes and MIRU into a new representation where traditional clustering methods apply (we use modified k-means clustering) without the 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 clusters with SpolDB4 families suggests dividing and merging some SpolDB4 families, while strongly validating others.

2 Background

In this study, we used two genotyping methods, spoligotyping and MIRU-VNTR typing, to cluster MTBC strains. We generated high-dimensional arrays to represent genotype information of MTBC strains. We mapped these high-dimensional arrays to two-dimensional space using multiway models and used score matrices of these models as input to k-means clustering of MTBC strains. We validated the clustering results using cluster stability and accuracy measures. In this section, we give a brief background on genotyping, multiway modeling and clustering of MTBC strains.

2.1 Spoligotyping

Spoligotyping is a DNA fingerprinting method that exploits the polymorphisms in the direct repeat (DR) region of the MTBC genome to distinguish between strains. The DR region is a polymorphic locus in the genome of MTBC which comprises of direct repeats (36 bp), separated by unique spacer sequences of 36 to 41 bp [9]. 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 contiguous spacers. To capture this evolution, 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.

2.2 MIRU-VNTR typing

MIRU is a homologous 46-100 bp DNA sequence dispersed within intergenic regions of MTBC, often as tandem repeats. Among the 41 identified mini-satellite regions on the MTBC genome, different subsets of size 12, 15 and 24 are proposed for standardization of MIRU genotyping [3]. In this study, we used 12-loci MIRU 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 loci.

2.3 Multiway analysis of biomarker tensor

The 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. This information is captured in a multi-dimensional array or tensor with three modes representing spoligotype deletions, MIRU patterns and strains. Mathematically, each strain is represented as the outer product of the binary spoligotype deletion vector and the MIRU loci, which results in a biomarker kernel matrix. Kernel matrices of the same size for each strain form the multiple-biomarker tensor. Multiway models analyze tensors by decomposing multiway arrays into two-way arrays. In this study, we use two common multiway models, PARAFAC and Tucker3. Dimensionality reduction on the tensor data using multiway models returns a score vector for each MTBC strain, which is used to measure similarities and corresponding distances between strains in a clustering algorithm. This is a key property of the algorithm since we don't know *a priori* how to measure evolutionary distance between isolates genotyped by MIRU typing and spoligotyping.

3 Methods

Clustering MTBC strains using multiple biomarkers consists of a sequence of steps. First, we generate a tensor with one mode representing the strains to be clustered, and two other modes representing the two biomarkers. Second, we apply multiway models on the strain mode of the tensor to get a score matrix of strains. Third, we use this score matrix to decide 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 clustering framework is outlined in Figure 1. We describe the steps of clustering framework in this section.

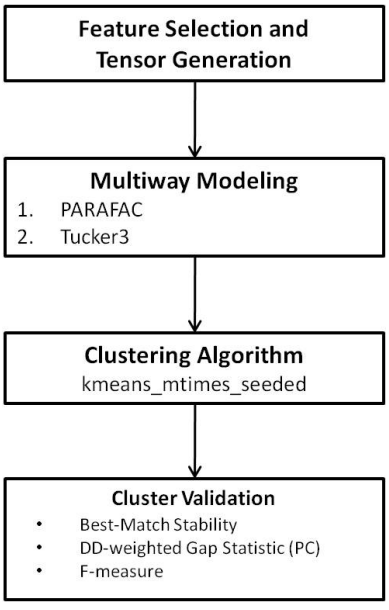


Fig. 1: Clustering framework of MTBC strains. High-dimensional genotype data is decomposed into two-dimensional arrays using multiway models, which are then used as input to `kmeans_mtimes_seeded` algorithm. Clusterings are validated using best-match stability. In case of a tie, DD-weighted gap statistic or F-measure is used to pick the number of clusters.

3.1 Datasets

The dataset comprises of 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 [4]. We also used the MIRU-VNTR*plus* dataset which is labeled with SpolDB4 lineages. The original SpolDB4 labeled dataset contains only spoligotypes. We found all occurrences of these spoligotypes in the CDC dataset. In this way we constructed a database with spoligotype and MIRU patterns, with major lineages as determined by CDC, and sublineages as given in SpolDB4. The numbers of strains for each major lineage are included in Table 1. We created 6 datasets from the CDC+MIRU-VNTR*plus* dataset, one for each major lineage, and divided them into sublineages.

Major lineage	# Strains	# Spoligo 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

TABLE 1: Number of strains in each major lineage of CDC+MIRU-VNTR*plus* dataset and number of spoligotype deletions identified by feature selection algorithm.

3.2 Feature Selection and Tensor Generation

3.2.1 Feature Selection

The spoligotype pattern captures the variability of the MTBC genome. Spoligotype is a 43-bit binary sequence, and according to hidden parent assumption, one or more contiguous spacers can be lost in a deletion event, but not gained [8]. Therefore, there are $\sum_{i=1}^{43} i = 946$ possible deletions in a spoligotype. Among these deletions, different spoligotype deletions were found effective in discrimination of MTBC strains. A set of 12 deletion sequences of spoligotypes found by Shabbeer et al. are proven to be good discriminator spacer deletions for major lineage classification [10]. Another set of 81 deletion sequences of spoligotypes found by Brudey et al. are proven to be good discriminator spacer deletions for SpolDB4 lineage classification [6].

We built a feature selection algorithm to find spacer deletions that are informative. Given a dataset, we first calculate the frequency f_i, $i = 1, .., 946$, of each deletion among the strains of the dataset. 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 in the middle 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 subsequence 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. Feature selection algorithm is summarized in Algorithm 1. Numbers of spoligotype deletions found informative for each major lineage are given in Table 1.

Algorithm 1 `FeatureSelection(StrainDataset)`

1: `Classify all possible 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.
2: `Remove uncommon deletions which are a supersequence of a common deletion`
3: **for** each uncommon deletion $u \in U$ **do**
4: **if** $\exists c \in C$ such that u is a supersequence of c **then**
5: Remove u from uncommon deletions: $U = U \setminus \{u\}$
6: **end if**
7: **end for**
8: Return uncommon deletion set U as the final feature set.

3.2.2 Multiple-Biomarker Tensor

The 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 $A(i, j, k)$ in the array corresponds to the number of repeats in MIRU pattern k of strain i with spoligotype deletion j. If spoligotype deletion j does not exist in strain i, then the tensor entry $A(i, j, .)$ is 0. Thus strain datasets are formed as *strain* $\times$ *spoligotype deletion* $\times$ *MIRU pattern* tensors, as shown in Figure 2. Generation of these multiple-biomarker tensors from the biomarker information of each strain is shown in Figure 3. We represent spoligotype deletions with $\vec{s}$, where $s_i \in \{0, 1\}$ and $i \in \{1, .., n\}$ where n is the number of informative spoligotype deletions found using feature selection algorithm. We represent 12-loci MIRU with $\vec{m}$, where $m_j \in \{0, .., 9, \geq 9\}$ and $j \in \{1, .., 12\}$.

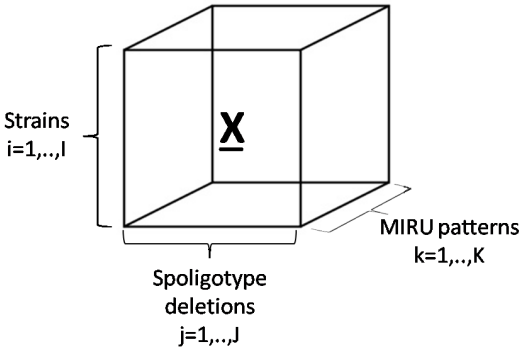


Fig. 2: *Strain* $\times$ *spoligotype deletion* $\times$ *MIRU pattern* tensor. Each entry $X(i, j, k)$ of the tensor represents the number of repeats in MIRU pattern k of strain i with spoligotype deletion j.

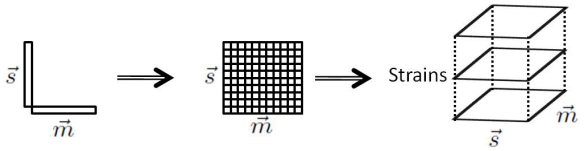


Fig. 3: Biomarker kernel matrix $\vec{s} \otimes \vec{m}$ for each strain forms multiple-biomarker tensor. $\vec{s}$ represents spoligotype deletions and $\vec{m}$ represents MIRU patterns.

3.3 Multiway modeling

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

3.3.1 Multiway models

We used PARAFAC and Tucker3 models to explain the tensor with high accuracy. Multiway modeling of multiple-biomarker tensors was carried out using *n-way Toolbox* of MATLAB by Bro et al. [11].

PARAFAC

PARAFAC is a generalization of SVD to multiway data [12], [13]. 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:

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

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. $\underline{\mathbf{E}} \in \mathbb{R}^{I \times J \times K}$ is the residual term containing all unexplained variation. A description of the PARAFAC model is shown in Figure 4.

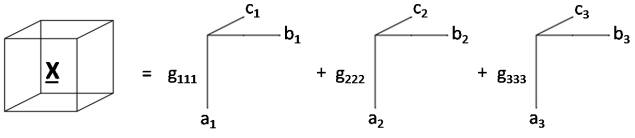


Fig. 4: PARAFAC model of a three-way tensor $\underline{\mathbf{X}}$. The tensor is modeled as a linear combination of rank-one tensors for each mode.

The PARAFAC model is symmetric in all modes and the number of components in each mode are the same [14]. The PARAFAC model is a simple model, which comes with a restriction 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 data returns the same results.

Tucker3

Tucker3 is an extension of bilinear factor analysis to multiway datasets [15]. 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:

$$\underline{\mathbf{X}}_{ijk} = \sum_{p=1}^{P} \sum_{q=1}^{Q} \sum_{r=1}^{R} \underline{\mathbf{G}}_{pqr} \mathbf{A}_{ip} \mathbf{B}_{jq} \mathbf{C}_{kr} + \underline{\mathbf{E}}_{ijk} \quad (2)$$

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 5.

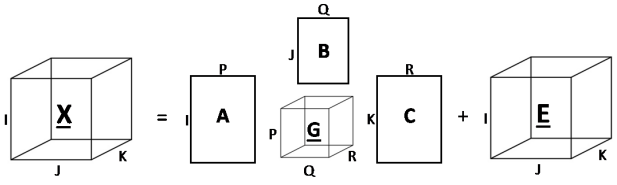


Fig. 5: Tucker3 model of a three-way tensor $\underline{\mathbf{X}}$ with (P, Q, R) components at each mode. The tensor is decomposed into component matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, core array $\underline{\mathbf{G}}$, and residual array $\underline{\mathbf{E}}$.

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 [16]. Therefore, the number of components for each mode can be different. This results in indeterminacy of the Tucker3 model, since it cannot determine component matrices uniquely.

3.3.2 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 [17]. The core consistency diagnostic measures the similarity of the core array $\underline{\mathbf{G}}$ of the model and the super-diagonal 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 an appropriate model [17]. The validated number of components along with core consistency values are shown in Table 2.

In order to determine the number of components of the Tucker3 model, we started with fitting a Tucker3 model with same number of components to the tensor. We picked the number of components that explains the variance of the data with close to 100% accuracy. Then we decreased the number of components until the most important factor combinations are found that explain over 90% of the variance of the data. The validated number of components along with the percentage of variance explained are shown in Table 2. Other methods for selecting the dimensions of a Tucker3 model include Difference in Fit (DIFFIT), deviance analysis and st-criterion based on convex hulls [14], [18], [19].

Major Lineage	PARAFAC		Tucker3	
	# Components	Core Consistency	# Components	Variance
M. africanum	3	94.79	[4 4 3]	95.66
M. bovis	2	100.00	[7 6 4]	95.05
East Asian (Beijing)	2	100.00	[3 4 2]	93.09
East-African Indian (CAS)	2	100.00	[11 10 4]	97.23
Indo-Oceanic	4	94.32	[15 13 5]	95.55
Euro-American	14	99.03	[14 13 5]	89.77

TABLE 2: Number of components used in PARAFAC and Tucker3 model to fit the tensors for the datasets to be clustered. We used core consistency diagnostic to validate PARAFAC models and percentage of variance explained by the model to validate Tucker3 models.

3.4 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 multiway 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. [20]. 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 2 summarizes the `getInitialCentroids` algorithm. 2) The local minima problem is partially solved by repeating `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 shown in algorithm 3. The `kmeans_mtimes_seeded` algorithm is more stable compared to the `k-means` algorithm, and produces more accurate results.

Algorithm 2 `getInitialCentroids(A,k)`

1: Pick the first centroid c_1 at random
2: **for** $i = 2$ to k **do**
3: Find $D(a)$, distance to closest centroid picked so far, for each data point $a \in$ A
4: Pick the data point a with maximum $D(a)$ as new centroid
$$c_i = \arg\max_a D(a)$$
5: Add c_i to the set of initial centroids
6: **end for**

Algorithm 3 `kmeans_mtimes_seeded(A,k,m)`

1: **for** $i = 1$ to m **do**
2: `initCentroids = getInitialCentroids(A,k)`
3: Apply k-means with `initCentroids`
4: Get the objective value of k-means run
5: **end for**
6: Pick the k-means run with minimum objective value

3.5 Cluster Validation

Clustering results for the MTBC strains are evaluated to determine the best choice for the number of clusters and compare it with existing sublineages using cluster validity indices. We used the best-match stability to pick the most stable clusterings. In case of a tie in average best-match stability, we used the DD-weighted gap statistic or F-measure for cluster validation [21].

3.5.1 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 [22]. Let the dataset be $X \in \mathbb{R}^{n\times p}$ 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:

$$W_k = \sum_{r=1}^{k} \frac{1}{2n_r} D_r$$

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 expected value under a sample of size n based on reference distribution. The optimal number of clusters is the value $\hat{k}$ for which $Gap_n(k)$ is maximized.

The reference distribution can be one of the 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 [22]. Computation of the gap statistic is summarized in algorithm 4.

Algorithm 4 `Gap Statistic(X,kmax,B)` [22]

1: Cluster the data X, varying the total number of clusters from k=1,2,..kmax, with corresponding W_k values.
2: Generate B reference datasets using the reference distribution, and find the corresponding W_{kb}^* values for b=1,2,..,B and k=1,2,..,kmax. The estimated gap statistic is found by:
$$Gap(k) = \frac{1}{B}\sum_{b=1}^{B} log(W_{kb}^*) - log(W_k)$$
3: Let $\bar{l} = \frac{1}{B}\sum_{b=1}^{B} log(W_{kb}^*)$ and compute the standard deviation:
$$sd_k = \left(\frac{1}{B}\sum_{b=1}^{B}\left(log\left(W_{kb}^* - \bar{l}\right)\right)^2\right)^{1/2}$$
and $s_k = sd_k\sqrt{(1+\frac{1}{B})}$. Choose the number of clusters $\hat{k}$ as the smallest k such that $Gap(k) \geq Gap(k+1) - s_{k+1}$.

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 gap statistic does not estimate the correct number of clusters for every case [23]. This may be because W_k increases as the number of data points increase. Hierarchical structure of the data may also cause problems. The data may be composed of nested clusters and gap statistic will be capturing only the minimum of these candidate number of clusters. Therefore, the gap statistic needs improvement. Yan et al. suggested a 2-step improvement to gap statistic, called the DD-weighted gap statistic [24]. 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 $\overline{W_k}$, 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 number of clusters is raised from k-1 to k. $D\overline{Gap}_n(k)$ is defined as

$$D\overline{Gap}_n(k) = \overline{Gap}_n(k) - \overline{Gap}_n(k-1) \quad (3)$$

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

$$DD\overline{Gap}_n(k) = D\overline{Gap}_n(k) - D\overline{Gap}_n(k+1) \quad (4)$$

From equations (3) and (4), $DD\overline{Gap}_n(k)$ is defined as:

$$DD\overline{Gap}_n(k) = 2\overline{Gap}_n(k) - \overline{Gap}_n(k-1) - \overline{Gap}_n(k+1) \quad (5)$$

$DD\overline{Gap}_n(k)$ is maximized when k is equal to 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 of the data. In such cases, multilayer analysis should be used instead of a single step procedure.

3.5.2 Best-Match Stability

The stability of a clustering is found 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 a clustering algorithm should produce clusterings that do not vary much for different subsamples [25]. Therefore, the algorithm is stable independent of input randomization. Several stability methods to estimate the correct number of clusters were proposed. Ben-Hur et al. suggested a stability-based model explorer algorithm, Lange et al. suggested a stability-based model order selection algorithm [26], [27]. We use best-match stability suggested by Hopcroft et al. [28]. The algorithm clusters the same data multiple times, and compares the reference cluster to alternate clusterings. We used 25 model clusterings to compare with the reference cluster. 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. 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 procedure is summarized in algorithm 5.

Algorithm 5 `BestMatchStability(`C^*`, Cref)`

k = number of clusters obtained from C^*
B = number of reference clusterings obtained from `Cref`
$Stability = zeros(k)$
for $i = 1$ to k **do**
  C = Strains in cluster i in clustering C^*
  for $j = 1$ to B **do**
    $\bigcup_{m=1}^{k} refC_m = Cref(j)$
    $best\_match(C, \bigcup_{m=1}^{k} refC_m) = \max_{m=1,..,k} match(C, refC_m)$
    where
    $$match(C, C^{'}) = \frac{\mid C \cap C^{'} \mid}{\max(\mid C \mid, \mid C^{'} \mid)}$$
    $Stability(i) = Stability(i) + best\_match$
  end for
end for
$Stability = Stability \,/\, B$
$AverageBestMatch = Sum(Stability) \,/\, k$
Return $Stability$ and $AverageBestMatch$

F-measure

The F-measure is a weighted combination of precision and recall of a clustering. We use the F-measure to evaluate how similar the tensor sublineages are to the SpolDB4 families. According to contingency table in Table 3, precision and recall are defined as follows:

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

TABLE 3: Contingency table. a is the number of data points that belong to same class and same cluster, b is the number of data points that belong to same class but different clusters, c is the number of data points that belong to different classes but same cluster, d is the number of data points that belong to different classes and different clusters. Given that there are n data points in the datasets, the following condition holds: $a + b + c + d = \binom{n}{2}$.

$$P = \frac{a}{a+c}$$
$$R = \frac{a}{a+b}$$

Different weights for precision and recall have been used to define F-measure, but the most common of all defines F-measure as the harmonic mean of precision and recall, as follows:

$$F = \frac{2PR}{P+R}$$

Since the F-measure combines precision and recall of clustering results, it has proven to be a successful metric.

4 Results

We subdivide each of the major lineages of MTBC into sublineages using multiple-biomarker tensors. For each major lineage,

we generated the multiple-biomarker tensor using spoligotypes and MIRUs and applied multiway models to identify putative sublineages of each major lineage. To evaluate the resulting clusters, we compare them with the published SpolDB4 families for each major lineage dataset. The results are summarized in Table 4. For each lineage, results show that the tensor approach finds highly stable sublineages (the best-match stability is $\geq$85%) and that the number of sublineages found using tensors is close but not always identical to the number of SpolDB4 families.

The F-measures range from 57% to 87% indicating that the sublineages found by the tensor only partially overlap with those of SpolDB4. Recall that the 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, thus in some cases the tensor divides SpolDB4 families due to difference in MIRU even if the spoligotypes match. In other cases, the tensor analysis merges together the SpolDB4 families because the collective spoligotypes and MIRU 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. 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.

4.1 Sublineage structure of *M. africanum*

The tensor methodology used Tucker3 to construct four distinct sublineages for *M. africanum*. Table 5 gives the stability of each sublineage and the correspondence between the tensor sublineages and the SpolDB4 families. The four 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 6. Figure 7 shows heat maps representing the spoligotype and MIRU signatures for each of the tensor sublineages with white indicating 0 probability and black indicating probability of 1.

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

TABLE 5: Confusion matrix for 64 distinct *M. africanum* strains showing the correspondence between the SpolDB4 families and tensor sublineages. The stability of each of the tensor sublineages is given in the second row. The clustering is validated by the best-match stability and DD-weighted gap statistic.

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 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_3 and AFRI_2 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 signature clearly distinguishes them especially in

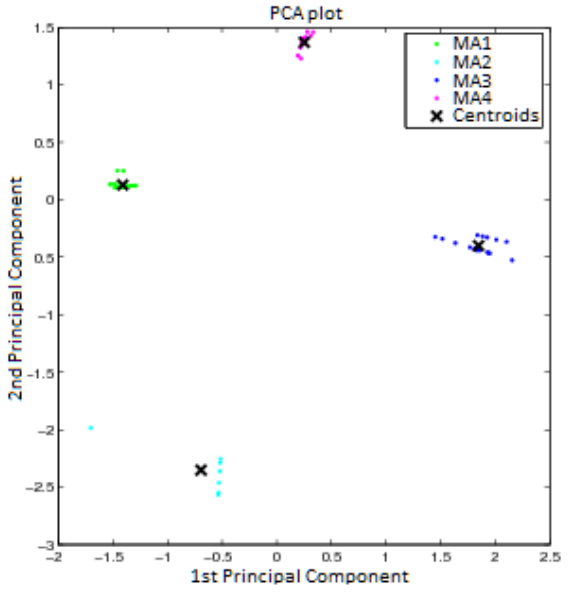


Fig. 6: Clustering plot of *M. africanum* strain dataset using Principal Component Analysis (PCA).

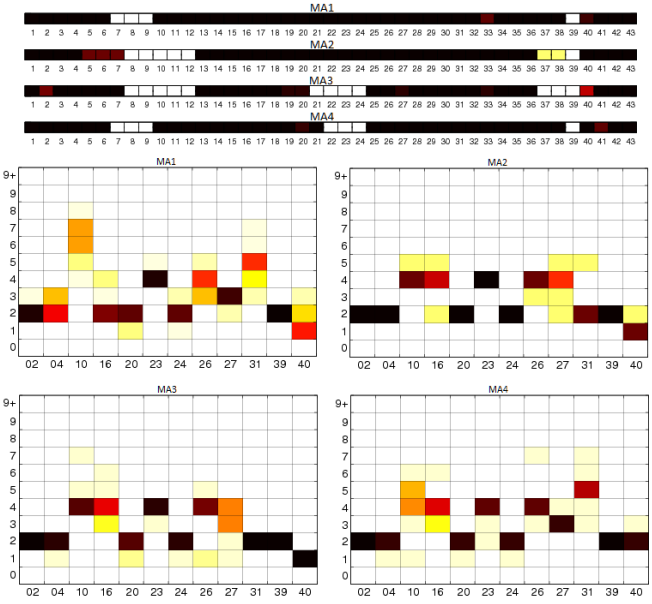


Fig. 7: Spoligotype and MIRU signatures of tensor sublineages of *M. africanum* strain dataset.

MIRU loci 10, 12 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_3) and MA3 (AFRI_2).

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 6 indicates the correspondence between the tensor sublineages and MIRU-VNTR*plus* labels. MA1 and MA4 clearly correspond to West African 2 and MA3 corresponds to West African 1. There is no data labeled by MIRU-VNTR*plus* in MA2, but we speculate that it is West African 1 since MA2 and MA3 have more closely related MIRU and spoligotype signatures.

Major Lineage	# SpolDB4 families	# Tensor sublineages	F-measure	Average best-match stability
M. africanum	4	4	0.66	1
M. bovis	5	3	0.71	1
East Asian (Beijing)	2	5	0.87	1
East-African Indian (CAS)	4	3	0.82	1
Indo-Oceanic	13	11	0.57	0.90
Euro-American	33	33	0.61	0.85

TABLE 4: Number of SpolDB4 families and number of tensor sublineages for each major lineage. F-measure and average best-match stability values assess the agreement of the sublineages to the SpolDB4 families and the certainty of tensor sublineages respectively.

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

TABLE 6: Confusion matrix for 64 distinct *M. africanum* strains showing the correspondence between the West African 1 and 2 sublineages and tensor sublineages. For data not from MIRU-VNTR*plus*, the lineage is indicated as unspecified.

4.2 Sublineage structure of *M. bovis*

The tensor methodology used PARAFAC to construct 3 sublineages for *M. bovis*, MB1, MB2 and MB3, while the dataset contains 5 SpolDB4 families, BOV, BOVIS1, BOVIS1_BCG, BOVIS2 and BOVIS3. Table 7 gives the correspondence between the tensor sublineages and the SpolDB4 families. All the clusters have perfect stability and are well distinguished in the PCA plot in Figure 8. Figure 9 shows heat maps representing the spoligotype and MIRU signatures of tensor sublineages. Much like the *M. africanum* SpolDB4 AFRI family, the BOV family defines a generic *M. bovis* that spreads across all three tensor sublineages. Disregarding BOV, MB1 consists of all of BOVIS1 and BOVIS1_BCG. Since BOVIS1_BCG is the attenuated bacillus Calmette-Guérin (BCG) vaccine strain, it is difficult to distinguish it from BOVIS1 using only MIRU and spoligotypes. Therefore, the merger of BOVIS1 and BOVIS1_BCG makes genetic sense. Disregarding BOV, the MB2 and MB3 sublineages exactly match the SpolDB4 families BOVIS3 and BOVIS2 respectively.

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

TABLE 7: Confusion matrix of *M. bovis* strain dataset clustered into 3 groups using PARAFAC. Correct labels are SpolDB4 labels on the rows, and tensor sublineages are represented by each column. The clustering is validated by the best-match Stability and F-measure.

4.3 Sublineage structure of East Asian (Beijing)

The tensor methodology used PARAFAC to construct five distinct sublineages for East Asian denoted B1 through B5. The variability in the spoligotypes of East Asian is limited to spacers 35 through 43 since all East Asian strains have spacers 1 to

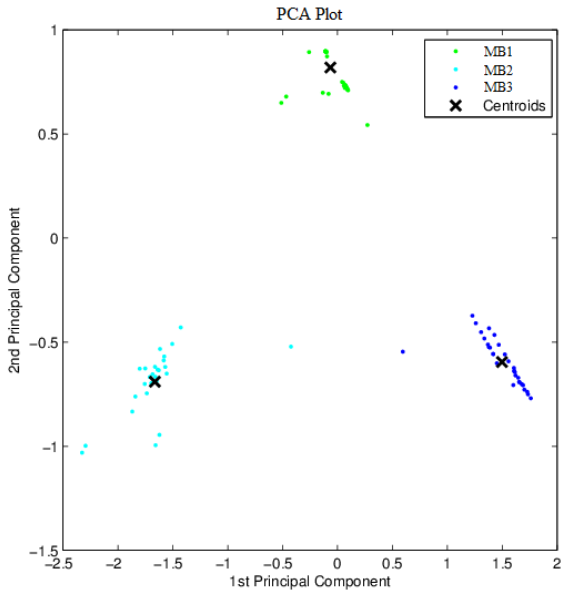


Fig. 8: Clustering plot of *M. bovis* strain dataset using Principal Component Analysis (PCA).

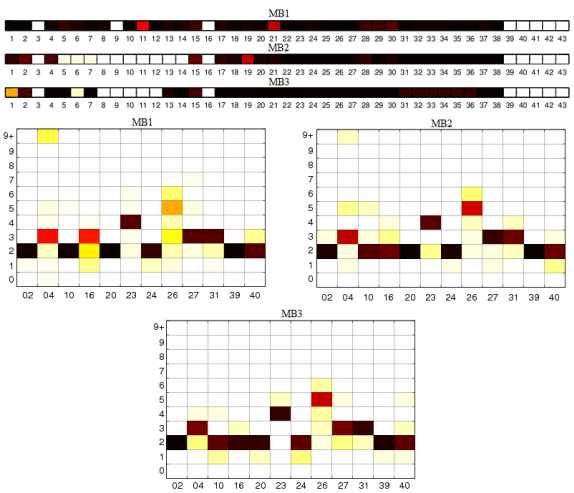


Fig. 9: Spoligotype and MIRU signatures of tensor sublineages of *M. bovis* strain dataset.

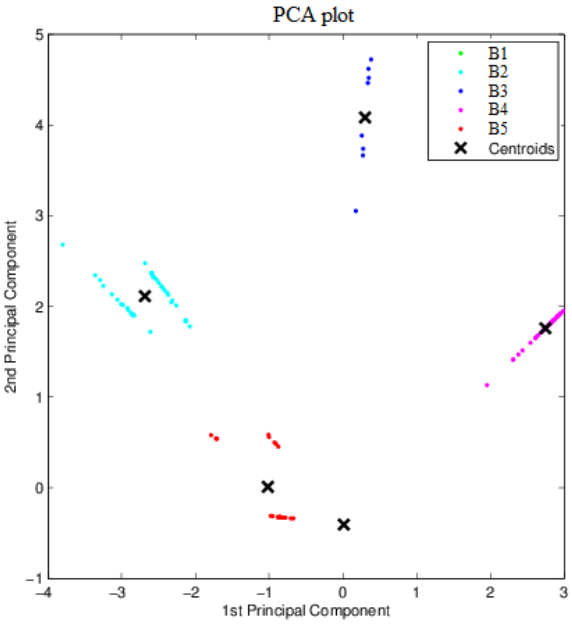


Fig. 10: Clustering plot of East Asian (Beijing) strain dataset using Principal Component Analysis (PCA).

34 absent. Since the SpolDB4 classification is based only on spoligotypes, the limited variability allows only two families, BEIJING and BEIJING-LIKE. Table 8 shows the correspondence between tensor sublineages and the SpolDB4 families. Clustering plot of tensor sublineages is shown in Figure 10. Heat maps representing the spoligotype and MIRU signatures of tensor sublineages are shown in Figure 11. The tensor cleanly subdivides BEIJING into three sublineages B1, B4 and B5 all with stability 1. Spoligotype signatures of these sublineages differ, and MIRU signature of sublineage B5 is clearly distinct in MIRU locus 40. The tensor subdivides the BEIJING-LIKE into sublineages B2 and B3 each with distinct spoligotype signatures. Thus the tensor strongly supports the existence of BEIJING and BEIJING-LIKE families, but also suggests that they can be further subdivided.

	B1	B2	B3	B4	B5
Stability	1	1	1	1	1
BEIJING	463	0	0	41	23
BEIJING-LIKE	0	36	8	0	0

TABLE 8: Confusion matrix of East Asian (Beijing) strain dataset clustered into 5 groups using PARAFAC. Correct labels are SpolDB4 labels on the rows, and tensor sublineages are represented by each column. The clustering is validated by the best-match Stability and DD-weighted gap statistic.

4.4 Sublineage structure of East-African Indian (CAS)

The tensor methodology used PARAFAC to construct three distinct sublineages for East-African Indian (also known as CAS) denoted C1, C2 and C3 while the dataset has four SpolDB4 lineages CAS, CAS1_DELHI, CAS1_KILI and CAS2. Table 9 shows the correspondence of tensor sublineages and SpolDB4 families. Figure 12 shows the clustering plot of tensor sublineages and Figure 13 shows spoligotype and MIRU signatures of tensor sublineages. All sublineages are highly stable with stability 1. Much like with AFRI and BOV, the generic CAS family was divided across C1, C2, and C3 sublineages.

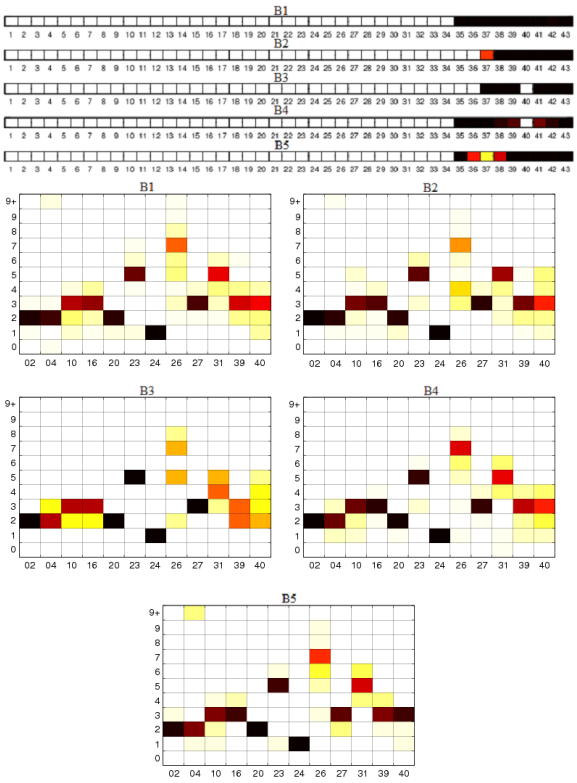


Fig. 11: Spoligotype and MIRU signatures of tensor sublineages of East Asian (Beijing) strain dataset.

Disregarding CAS, C1 only contains CAS1_DELHI and C3 only contains CAS2. C2 contains all of CAS1_KILI. C2 also contains 6 CAS1_DELHI strains, but the vast majority (327 strains) of CAS1_DELHI fall in C1. Variabilities in MIRU loci 10, 26, and 40 are key to defining differences in the sublineages along with distinct deletion patterns in the spoligotypes.

	C1	C2	C3
Stability	1	1	1
CAS	58	43	6
CAS1_DELHI	327	6	0
CAS1_KILI	0	23	0
CAS2	0	0	45

TABLE 9: Confusion matrix of East-African Indian (CAS) strain dataset clustered into 3 groups using PARAFAC. Correct labels are SpolDB4 labels on the rows, and tensor sublineages are represented by each column. The clustering is validated by the best-match stability and DD-weighted gap statistic.

4.5 Sublineage structure of Indo-Oceanic

The tensor methodology used PARAFAC to construct eleven distinct sublineages for Indo-Oceanic denoted IO1 to IO11 while the dataset has thirteen SpolDB4 lineages. Table 10 shows the correspondence of tensor sublineages and SpolDB4 families. Figure 14 shows the clustering plot of tensor sublineages and Figure 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 IO2 and IO5. The small MANU1 family also spreads across four sublineages. The existence of the MANU1

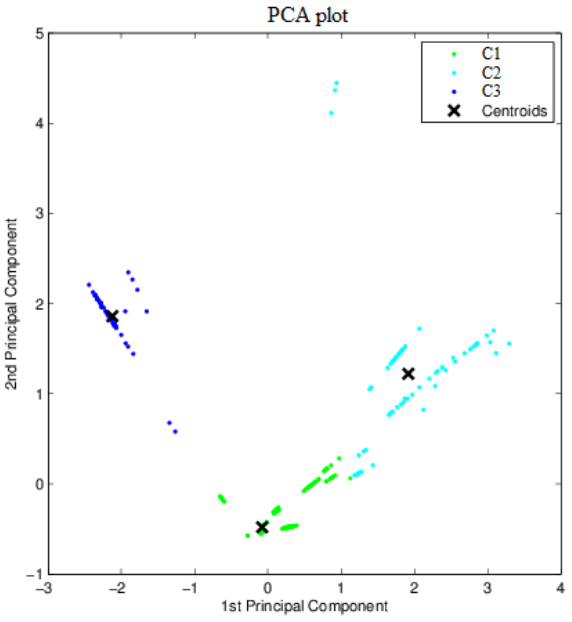


Fig. 12: Clustering plot of East-African Indian (CAS) strain dataset using Principal Component Analysis (PCA).

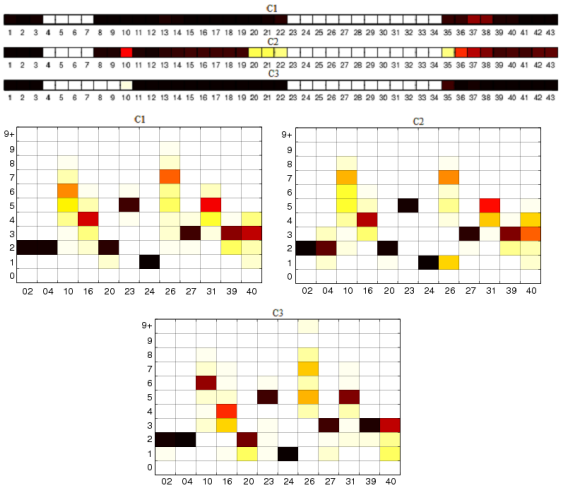


Fig. 13: Spoligotype and MIRU signatures of tensor sublineages of East-African Indian (CAS) strain dataset.

family has not been well established by other biomarkers. Disregarding these two troubling families, the tensor sublineages correspond closely to the SpolDB4 families. Specifically, the mapping between the most stable clusters (with sublineages stability) and the families are IO1 (.99) equals EAI3_IND, IO2 (1) equals ZERO, IO3 (.99) equals EAI2_NTB, IO4 (.98) equals a subset of EAI5, IO9 (.97) equals some EAI5 plus all of EAI8_MDG and some of EAI1_SOM, IO11 (.94) contains the vast majority of EAI1_SOM and EAI6_BDG1, and some of EAI5, and IO7 (.79) equals EAI4_VNM and EAI. EAI2_MANILLA is subdivided into three sublineages: IO8 (1) consisting of 241 strains, IO5 (.81) with 24 strains, and IO10 (.69) with 11 strains. 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 general sublineage that is further subdivided. Distinct patterns are observable in the spoligotype and MIRU signatures for most of the lineages.

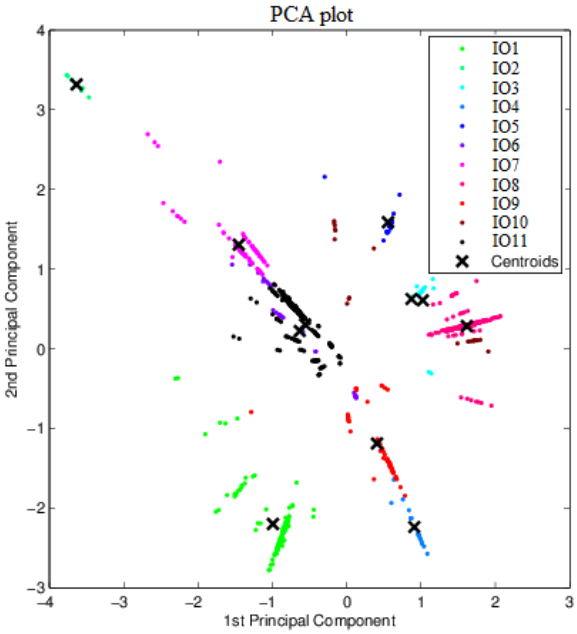


Fig. 14: Clustering plot of Indo-Oceanic strain dataset using Principal Component Analysis (PCA).

4.6 Sublineage structure of Euro-American

We used Tucker3 to find 33 sublineages for Euro-American denoted E1 to E33, the same number as the dataset which has 33 SpolDB4 lineages. Table 11 shows the correspondence of tensor sublineages and SpolDB4 families. Figure 16 shows the clustering plot of tensor sublineages and Figure 17 shows the spoligotype and MIRU signatures of tensor sublineages.

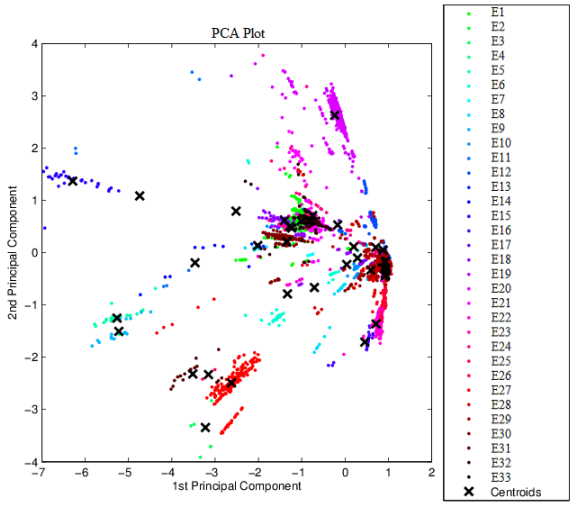


Fig. 16: Clustering plot of Euro-American strain dataset using Principal Component Analysis (PCA).

Strains belonging to families H2, H37Rv, H4, LAM12_MAD1, T1 (Tuscany variant), T1_RUS2, T4, T5_MAD2 and T5_RUS1 are clustered in tensor sublineages E15, E24, E12, E8, E18, E6, E29, E29 and E18 respectively. In contrast, the T1 family, an ancestor strain family, is distributed across 25 tensor sublineages, with most of the T1 strains in E29. 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, E14 and E25

	IO1	IO2	IO3	IO4	IO5	IO6	IO7	IO8	IO9	IO10	IO11
Stability	0.99	1	0.99	0.98	0.81	0.76	0.79	1	0.97	0.69	0.94
EAI	0	0	0	0	0	0	6	0	0	0	0
EAI1	0	0	0	0	0	0	0	0	0	0	2
EAI1_SOM	0	0	0	0	0	4	0	0	8	0	105
EAI2_MANILLA	0	0	0	0	24	0	0	241	0	11	0
EAI2_NTB	0	0	15	0	0	0	0	0	0	0	0
EAI3_IND	105	0	0	0	0	0	0	0	0	0	0
EAI4_VNM	0	0	0	0	1	0	44	0	0	0	0
EAI5	23	0	2	17	0	25	28	10	45	7	235
EAI6_BGD1	0	0	0	0	0	1	0	0	0	0	42
EAI8_MDG	0	0	0	0	0	0	0	0	4	0	0
MANU1	0	0	0	0	0	1	2	5	0	0	1
MICROTI	0	0	0	0	0	0	0	0	0	3	0
ZERO	0	6	0	0	0	0	0	0	0	0	0

TABLE 10: Confusion matrix of Indo-Oceanic strain dataset clustered into 11 groups using PARAFAC. Correct labels are SpolDB4 labels on the rows, and tensor sublineages are represented by each column. The clustering is validated by the best-match stability and F-measure.

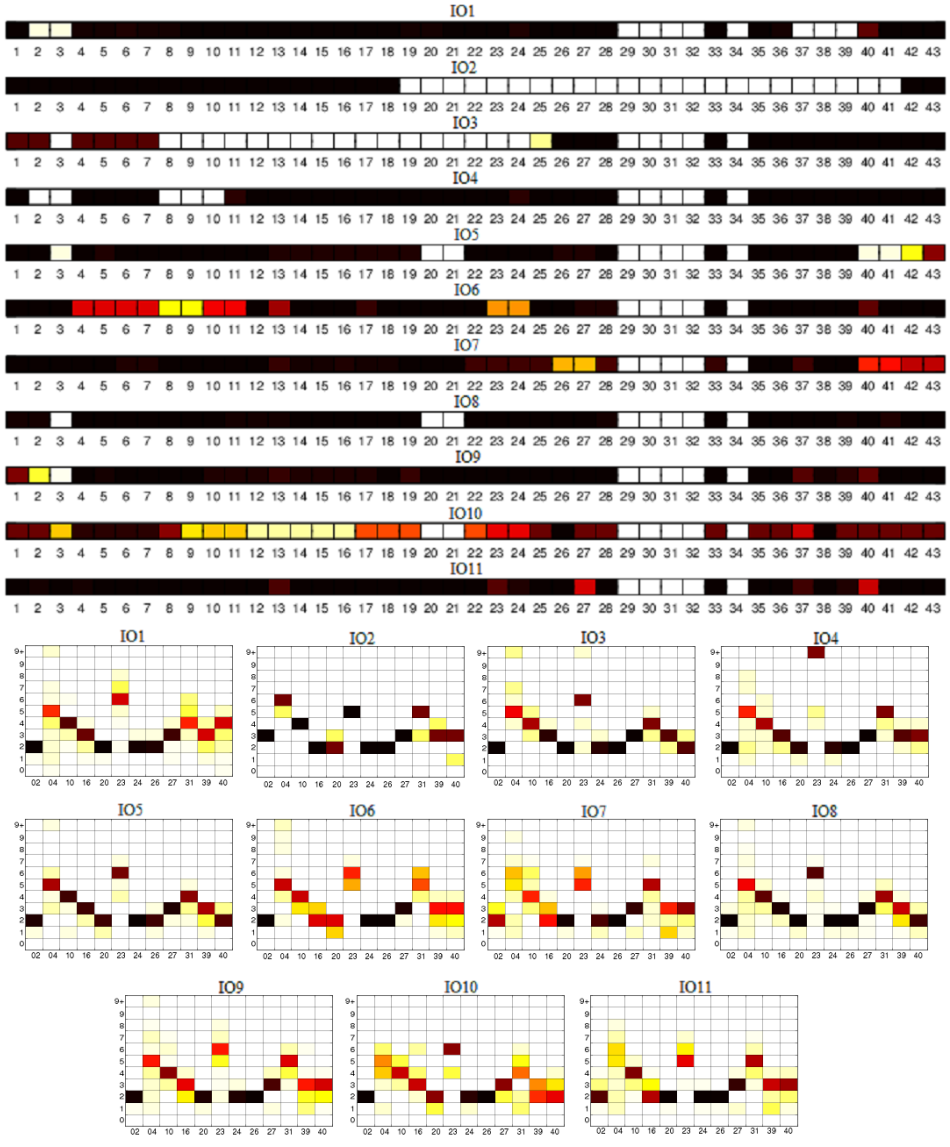
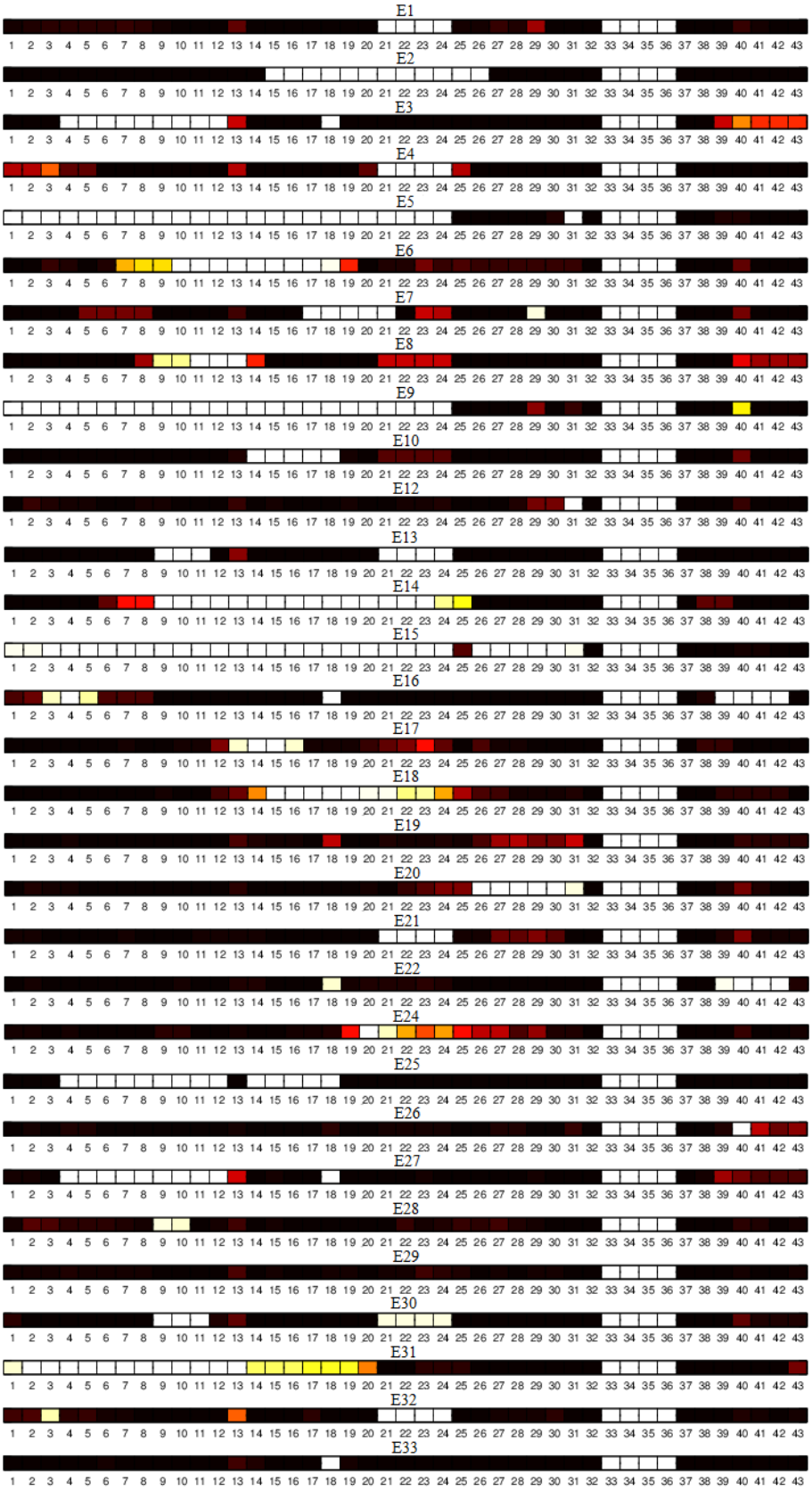


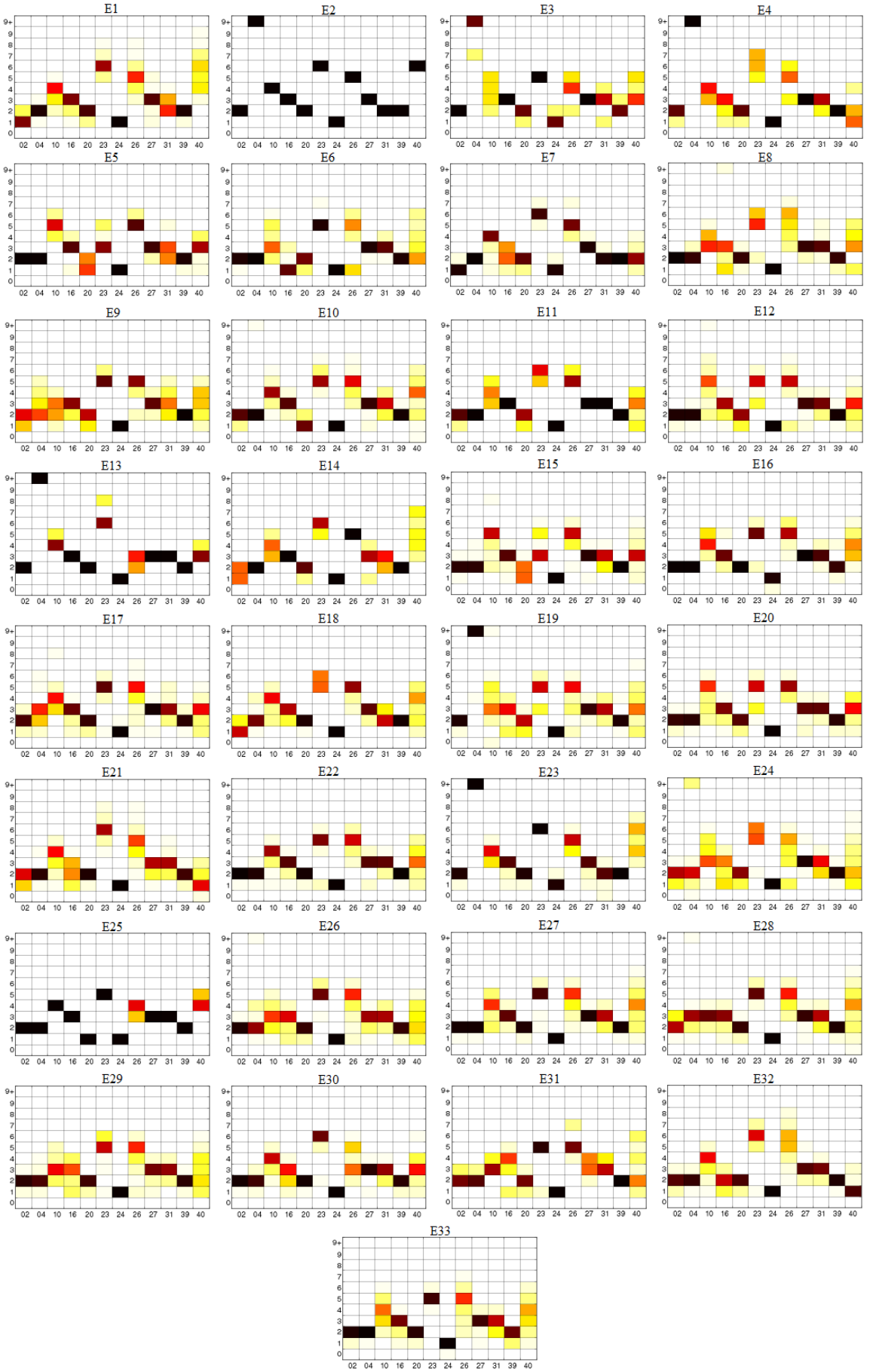
Fig. 15: Spoligotype and MIRU signatures of tensor sublineages of Indo-Oceanic strains.

	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	E11	E12	E13	E14	E15	E16	E17	E18	E19	E20	E21	E22	E23	E24	E25	E26	E27	E28	E29	E30	E31	E32	E33
Stability	0.97	1	0.96	0.90	0.90	0.79	0.38	0.97	0.77	0.93	0.71	0.89	0.97	0.60	0.89	1	0.93	0.85	0.90	0.99	0.95	0.98	0.97	0.58	0.78	0.94	0.76	0.96	0.83	0.97	0.63	0.97	0.49
H1	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	1	4	240	0	0	0	1	0	0	0	0	0	0	0	0	0
H2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
H3	0	0	0	0	25	2	0	0	1	0	3	377	0	0	3	0	0	1	12	46	0	0	1	1	0	19	0	1	6	0	0	0	2
H37Rv	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0	0	0	0	0	0	0	0
H4	0	0	0	0	0	0	0	0	0	0	0	59	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
LAM1	2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	80	0
LAM10_CAM	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	3	0	0	29	0	1	0	0
LAM11_ZWE	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	24	0	0	0	0	0	0	0	0	0	0	3	0
LAM12_MAD1	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
LAM2	1	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	87	0
LAM3	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	188	0	0	0
LAM4	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	27	0	2	0	0	1	0	0	0	0	0	3	0
LAM5	30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	23	0
LAM6	41	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	13	0	1	0	0	0	0	0	0	0	0	0	0
LAM7_TUR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0
LAM8	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0
LAM9	184	0	0	3	0	1	0	0	0	5	0	0	0	0	0	0	2	0	1	0	178	1	23	0	0	2	3	0	1	2	0	24	0
MANU2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
S	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	177	0	0	0	0	0
T1	7	1	0	2	0	0	17	16	3	33	3	0	0	8	1	0	27	35	20	8	5	14	0	22	3	45	0	9	906	12	17	2	10
T1 (Tuscany variant)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
T1_RUS2	0	0	0	0	0	18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
T2	0	0	0	0	0	0	4	3	11	7	1	0	0	0	0	0	0	0	1	1	1	0	0	4	0	173	0	2	18	0	0	0	0
T3	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	5	0	0	92	0	0	0	0
T3-OSA	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0	0
T3_ETH	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
T4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	0	0	0	0
T5	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	10	2	0	4	0	0	0	0	0	2	0	0	33	0	0	0	0
T5_MAD2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	14	0	0	0	0
T5_RUS1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
X1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	15	0	0	4	0	0	0	16	2	0	29	0	0	0	327
X2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	27	0	0	0	0	0	182	0	0	0	0	0	0	0	0	0	0	0
X3	0	0	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	188	0	0	0	0	0	0

TABLE 11: Confusion matrix of Euro-American strain dataset clustered into 33 groups using Tucker3. Correct labels are SpolDB4 labels on the rows, and tensor sublineages are represented by each column. The clustering is validated by Best-Match Stability and F-measure.



a) Spoligotype signature of tensor sublineages of Euro-American strains.



b) MIRU signature of tensor sublineages of Euro-American strains.

Figure 17: Spoligotype and MIRU signatures of tensor sublineages of Euro-American strains.

include T1 strains only. In addition to common spacer deletions of Euro-American strains, E2 lacks spacers 15 through 26, E14 lacks spacers 9 through 23 and E25 lacks spacers 3 through 12 and 14 through 18. This sublineage classification further subdivides the poorly-defined ancestor T1 family. Strains of LAM families on the other hand are grouped together in tensor sublineages E1 and E21. Prior studies have found that LAM Rio strains identified by SNPs are found in multiple SpolDB4 lineages [29]. Therefore, it is not surprising that use of the multiple biomarkers leads to subdivision or merging of some SpolDB4 families.

5 Conclusion

We developed a clustering framework which groups MTBC strains based on their spoligotype and MIRU information via multiple-biomarker tensors. We generated multiple-biomarker tensors for representation of high-dimensional biomarker information and used multiway models for dimensionality reduction. The multiway representation determines a transformation of the data that captures the similarities and differences between strains based on two distinct biomarkers. We clustered MTBC strains based on 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 lead to coherent sublineages of major lineages with clear and distinctive spoligotype and MIRU signatures.

The 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. We can use spoligotype and MIRU to group MTBC strains and compare them to labels derived from SNPs. Representation of MTBC genotype via multiple-biomarker tensors can also be extended to include 15-loci and 24-loci MIRU pattern. Moreover, more biomarkers can be used in the MTBC strain genotype representation. We can extend multiple-biomarker tensors and add a new mode for each biomarker added to genotype representation of strains, such as RFLP. This would be a major advancement because there is no way to define a similarity measure between RFLPs of strains other than determining whether or not the patterns match exactly. Addition of new biomarkers will increase the number of modes of the multiple-biomarker tensor, while the multiway analysis methods remain the same.

Future work will involve using various biomarkers to group MTBC strains. Multiple-biomarker tensors with spoligotype, MIRU patterns, and RFLP in modes may lead to a clustering of MTBC strains which is comparable with lineages identified on the basis of SNPs. This flexible representation should enable identification of subgroups of MTBC strains based on nucleotide sequences in one of the modes. Since many subfamilies are clearly known and more biomarkers are being developed, the multiple-biomarker tensor can be used in supervised and even semi-supervised classification to build reliable classifiers of MTBC sublineages and can be used to enhance TB control, epidemiology and research.

Acknowledgments

This work was made possible by Dr. Lauren Cowan and Dr. Jeff Driscoll of the Centers for Disease Control and Prevention. This work was supported by NIH R01LM009731.

References

[1] S. Gagneux *et al.*, "Variable host-pathogen compatibility in *Mycobacterium tuberculosis*," *PNAS*, vol. 103, no. 8, pp. 2869–2873, 2006.

[2] 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.

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

[4] M. Aminian, A. Shabbeer, and K. P. Bennett, "A conformal Bayesian network for classification of *Mycobacterium tuberculosis* complex lineages," *BMC Bioinformatics*, vol. 11, no. Suppl 3, p. S4, 2010.

[5] S. Ferdinand *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.

[6] K. Brudey *et al.*, "*Mycobacterium tuberculosis* complex genetic diversity: mining the fourth international spoligotyping database (SpolDB4) for classification, population genetics and epidemiology," *BMC Microbiology*, vol. 6, p. 23, 2006.

[7] B. Asiimwe, "Molecular characterization of *Mycobacterium tuberculosis* complex in Kampala, Uganda," Ph.D. dissertation, Makerere University, 2008.

[8] I. Vitol, J. Driscoll, B. Kreiswirth, N. Kurepina, and K. P. Bennett, "Identifying *Mycobacterium tuberculosis* complex strain families using spoligotypes," *Infection, Genetics and Evolution*, vol. 6, no. 6, pp. 491 – 504, 2006.

[9] J. Kamerbeek *et al.*, "Simultaneous detection and strain differentiation of *Mycobacterium tuberculosis* for diagnosis and epidemiology," *Journal of Clinical Microbiology*, vol. 35, no. 4, pp. 907–914, 1997.

[10] A. Shabbeer, L. Cowan, J. R. Driscoll, C. Ozcaglar, S. L. Vandenberg, B. Yener, K. P. Bennett, "TB-Lineage: an online tool for classification and analysis of strains of *Mycobacterium tuberculosis* complex," 2010, unpublished manuscript.

[11] 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.

[12] 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.

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

[14] ——, *Applied Multiway Data Analysis*. Wiley, 2008.

[15] L. R. Tucker, "Some mathematical notes on three-mode factor analysis," *Psychometrika*, vol. 31, no. 3, pp. 279–311, 1966.

[16] E. Acar and B. Yener, "Unsupervised multiway data analysis: A literature survey," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 1, pp. 6–20, 2009.

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

[18] M. E. Timmerman and H. A. L. Kiers, "Three-mode principal components analysis: Choosing the numbers of components and sensitivity to local optima," *British Journal of Mathematical and Statistical Psychology*, vol. 53, no. 1, pp. 1–16, 2000.

[19] E. Ceulemans and H. A. L. Kiers, "Selecting among three-mode principal component models of different types and complexities: A numerical convex hull based method." *British Journal of Mathematical and Statistical Psychology*, vol. 59, no. 1, pp. 133 – 150, 2006.

[20] D. Arthur and S. Vassilvitskii, "K-means++: The advantages of careful seeding," in *SODA '07: Proceedings of the eighteenth annual ACM-SIAM symposium on discrete algorithms*. Society for Industrial and Applied Mathematics, 2007, pp. 1027–1035.

[21] M. Halkidi, Y. Batistakis, and M. Vazirgiannis, "Cluster validity methods: part I," *SIGMOD Rec.*, vol. 31, no. 2, pp. 40–45, 2002.
[22] R. Tibshirani, G. Walther, and T. Hastie, "Estimating the Number of Clusters in a Dataset via the Gap Statistic," vol. 63, pp. 411–423, 2000.
[23] 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.
[24] M. Yan and K. Ye, "Determining the number of clusters using the weighted gap statistic," *Biometrics*, vol. 63, no. 4, pp. 1031–7, 2007.
[25] S. Ben-David *et al.*, "A Sober Look at Clustering Stability," in *COLT*, 2006, pp. 5–19.
[26] G. I. Ben-Hur A, Elisseeff A, "A stability based method for discovering structure in clustered data," *Pacific Symposium on Biocomputing*, vol. 7, pp. 6–17, 2002.
[27] T. Lange *et al.*, "Stability-based validation of clustering solutions," *Neural Computation*, vol. 16, no. 6, pp. 1299–1323, 2004.
[28] J. Hopcroft *et al.*, "Natural communities in large linked networks," in *KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2003, pp. 541–546.
[29] A. L. Gibson *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. Microbiol.*, vol. 46, no. 4, pp. 1259–1267, 2008.