University of Nevada
Reno

A Fast-Graph Approach to Modeling Similarity of Whole Genomes

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Computer Science and Engineering

by

Adrienne E. Breland

Dr. Frederick C. Harris, Jr./Dissertation Co-Advisor
Dr. Karen A. Schlauch/Dissertation Co-Advisor

August, 2011

Acknowledgments

This dissertation is dedicated to my father, Dr. Albert Breland Jr., because of his unrelenting support for the education of his children. All future benefits I receive from completing a PhD go to my son, Reed.

I would also like to thank my outstanding co-advisors Dr. Karen Schlauch and Dr. Frederick C. Harris Jr. It's hard to believe my luck in having both of them advise me during my graduate studies at UNR. They have shown true concern for my education, my future and my well being. Thank you both so much.

I am also grateful to my other committee members: Dr. Monica Nicolescu, Dr. Mehmet Gunes and Dr. John Cushman. Each has provided extremely valuable insight and guidance throughout this endeavor.

Thanks as well to my lab mates; Roger Hoang, Cody White, Joe Mahsman, Joe Vesco, Jamie Hobel, Anne Paine and Josh Hegie. Your humor and help with computing is truly appreciated.

August 2011

Abstract

As increasing numbers of closely related genomic sequences become available, the need to develop methods for detecting fine differences among them also grows apparent. Several calls have been made for improved algorithms to exploit the wealth of pathogenic viral and bacterial sequence data that are rapidly becoming available to researchers. The first stage of our research addresses the computational limitations associated with whole-genome comparisons of large numbers of subspecies sequences. We investigate the potential for the use of fast, word-based comparative measures to approximate computationally expensive, full alignment comparison methods.

Recent advances in next generation sequencing are providing a number of large whole-genome sequence datasets stemming from globally distributed disease occurrences. This offers an unprecedented opportunity for epidemiological studies and the development of computationally efficient, robust tools for such studies. In the second stage of our research, we present an approach that enables a quick, effective, and robust epidemiological analysis of large whole-genome datasets. We then apply our method to a complex dataset of over $4,200$ globally sampled *Influenza A virus* isolates from multiple host types, subtypes and years. These sequences are compared using an alignment-free method that runs in linear-time. These comparisons enable us to build 2-dimensional graphs that represent the relationships between sequences, where sequences are viewed as vertices, and high-degree sequence similarity as edges. These graphs prove useful, as they are able to model potential disease transmission paths when applied to viral sequences. Mixing patterns are then used to study the occurrence and patterns of edges between different types of sequence groups, such as the host type and year of collection, to better understand the potential of genotypic transfer between sequence groups.

THE GRADUATE SCHOOL

We recommend that the dissertation
prepared under our supervision by

ADRIENNE E. BRELAND

entitled

A Fast-Graph Approach To Modeling Similarity Of Whole Genomes

be accepted in partial fulfillment of the
requirements for the degree of

DOCTOR OF PHILOSOPHY

Dr. Karen Schlauch, Co-Advisor

Dr. Frederick C. Harris, Jr., Co-Advisor

Dr. Monica Nicolescu, Committee Member

Dr. Mehmet Gunes, Committee Member

Dr. John Cushman, Graduate School Representative

Marsha H. Read, Ph. D., Associate Dean, Graduate School

August, 2011

Contents

List of Figures

List of Tables

Chapter 1

Introduction

As the availability of public whole genome sequences increases, a demand to develop methods that detect fine differences among these sequences grows as well. Several calls have been made for improved algorithms to exploit the wealth of pathogenic viral and bacterial sequence data [9, 30]. Whereas these types of data sets continue to expand, algorithms to compare many long sequences are still lacking [9, 30]. Comparing whole-genomes, as opposed to shorter sequence segments (such as selected genes of interest), is a robust comparative approach, using all available genetic information of the organism. Comparative conclusions are clearly dependent upon the genomic subsequences selected [83].

Computing sequence alignment scores using traditional dynamic programming can yield an optimal solution (the best possible alignment); however, this computation runs on the order of $O(N^2)$ in time where N is equal to sequence length. Thus, dynamic programming (DP) is not a practical approach to compare sequences of most longer whole genomes. For example, the *Eschericia coli* genome contains approximately five million base pairs. Comparing two of these sequences using a standard DP matrix would require $2.5x10^{13}$ memory locations and each would need to be visited at least once.

The first stage of our research addresses the computational limitations associated with whole-genome comparisons of subspecies sequences. We investigate and determine computationally efficient and accurate comparison methods that evaluate entire databases of viral subspecies whole-genomes. These methods include alignment-free,

k-mer based measures that run in linear-time. The goal is to find a reliable alignment-free replacement to alignment-based comparisons, which are computationally expensive. We assess the accuracy of each method via correlation of alignment-free distance scores with pairwise alignment scores generated by the popular multiple sequence alignment program ClustalW.

Genomic sequences, being the "molecules as documents of evolutionary history" [92], have proven integral to research involving the transmission of subspecies pathogens. Subspecies comparisons can enable insight into viral forensics and disease transmission patterns. In the second stage of our research, we describe a graphical approach to examine disease transmission. This entails comparing large numbers of whole genomes using a rapid, alignment-free algorithm and then creating a graph based upon these comparative measures. We apply our methodology to a large set of subspecies, *Influenza A virus* whole genome sequences, which are publicly available from the Influenza Virus Resource database [2]. We are then able to examine the possible transfer of genotypes across species and among virus collection years. We also compute the probabilities that viral isolate sequences across species and year groups in the given dataset are very similar.

There are several novel aspects to our research, to the best of our knowledge: A) ours is the first study that assesses the accuracy of using k-mer based comparisons for large, subspecies datasets, B) we have not encountered the creation of complex graphs from genomic sequences, where edges are drawn based upon sequence similarity scores and user-defined thresholds, C) the k-deep prefix tree algorithm, described in Chapter 6, and D) a word-based distance metric for comparing genomic sequences presented in Section 7.10.

In Chapter 2, we provide a review for computer scientists about the motivation behind genomic sequence comparisons. Chapter 3 provides the motivation for the proposed research. We also discuss the magnitude of data available and some limiting computational bottlenecks. A review of word-based comparative measures and why they are favorable for use in comparative algorithms are presented in Chapter 4.

Chapter 5 summarizes and outlines the research goals, and Chapter 6 describes some preliminary work regarding an efficient algorithm and data structure that might be applied to k-mer based comparisons. In Chapter 7, we test several k-mer based comparative approaches on viral datasets and select the most accurate method. In Chapter 8, we describe a methodological approach to build a graph from an *Influenza A virus* whole-genome data set. Chapter 8 also includes an analysis of the graph with respect to edges found between vertices collected from different host types and time points.

Chapter 2

Some Genomics for Computer Scientists

2.1 What is a genome to a computer scientist?

A genome contains the complete genetic material of an organism. DNA and RNA stand for deoxyribonucleic acid and ribonucleic acid molecules. These are long-stranded molecules composed of only four types of bases; Adenine, Cytosine, Guanine and Thymine (DNA) or Uracil (RNA) (A, C, G, T/U). Most organisms contain double-stranded DNA molecules, as shown in Figure 2.1; however, a group of viruses, referred to as RNA viruses, contain single-stranded RNA genomes. In RNA, every Thymine base is replaced by a Uracil. Either way, genomes are constructed from only four types of molecules rendering a small alphabet {A,C,G,T/U} with which all simple instructions must be encoded.

A computer scientist working with genomic sequences will find data in the form of long (very long) text strings (Figure 2.2). Computations generally involve sequence comparisons and/or pattern finding. New and diverse algorithms to achieve these tasks are published regularly in bioinformatics journals.

The image in Figure 2.2 is a piece of the code for human chromosome 21 which is 49,691,432 base pairs (bp) or characters long. Because of their length, genomic sequences and their analysis require notable amounts of memory, high power computing, and algorithms that are efficient in computational time and storage space. The digital format of a genomic sequence is simply a string; thus, many tools from

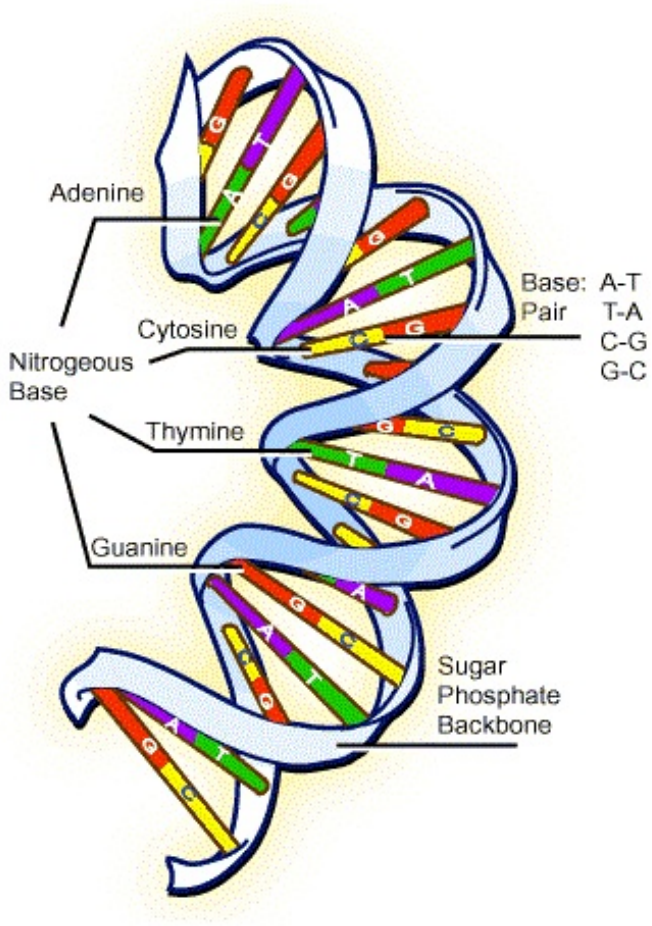

Figure 2.1: Depiction of a DNA molecule, from [71]

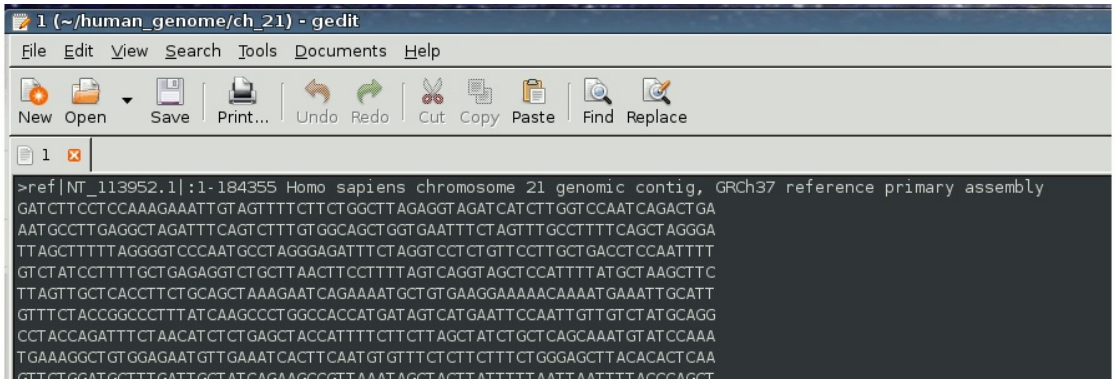

Figure 2.2: Nucleotide sequence string

string processing can be leveraged to analyze these sequences.

Genomic sequences contain many genes that can affect the on/off switch of many other genes, yielding unknown or unstudied sequence regions. Understanding how these genes interact with external, environmental and/or cellular factors is even more complicated. For example, epigenetics, post-transcriptional regulation and the histone code refer to changes in gene expression not dictated by the simple DNA code. Even locating all of the genes in the human genome is still a work in progress. Inter-genic regions are also still being explored. For example, about 0.3% to 1% of the human genome is composed of conserved non-genic sequences that are shared among all mammals [22]. Non-genic or inter-genic regions are those regions of DNA that do not encode for proteins; they are not genes. The fact that some of these regions are conserved indicates that they serve a crucial function; however, these functions often remain unclear.

So, what is a genome to a computer scientist? It is a long text string of only four letters that forms the basis of instructions for important functions like causing or preventing cancer, Parkinson's disease, or the virulence of the bubonic plague. Thanks to a long history of biochemical research and a shorter history of computational analysis, it is a code that is being deciphered rapidly.

2.2 Why compare genomic sequences?

Most computational analysis of genomes involves some sort of string and or substring comparisons. To create a useful algorithm, it is important to understand why these comparisons are made and what kind of knowledge can be gained. A famous essay written by the evolutionary biologist Theodisius Dobzhansky was titled "Nothing in Biology Makes Sense Except in the Light of Evolution" [16]. Genomic sequences differ from one another because they evolve. This means that, in the most basic sense, they change over generations. Computationally, the implication is that at any point in time, each organism on earth is represented by a unique string identifier. Furthermore, that string is the ultimate record of an organism's evolutionary history [87]. In

1965, Emile Zuckerkandl and Linus Pauling accurately predicted that DNA would be "molecules as documents of evolutionary history" [92].

When comparing genome strings, the basic assumption is that the more similar the strings, the more related the organisms. Sequences change through mutations which might accumulate over time. In general, types of mutations include point mutations, which are changes in single base pairs, or insertions and deletions, which involve the insertion or removal of small contiguous stretches. Point mutations occur during DNA replication or can be induced by mutagens. Insertions are usually the result of transposable elements, which are portions of DNA that might jump from one region to another. In these cases, deletions also occur upon the removal of an inserted region and potentially some of the surrounding regions as well. Insertions and deletions might occur during replication as well.

Mutation rates attributable to replication errors were examined in rats and mice [48]. In mammals, DNA in male germlines undergoes more cell division than female germlines. This accounted for a male-to-female sex bias in mutation rates of approximately 2:1.

Some, but not all errors are terminal. Genomic changes can cause an organism to die, to have reduced fitness and produce less offspring, to experience no change, or to gain an adaptive advantage. Evolutionary selection describes this concept. Evolutionary selection is broken down into purifying (negative) selection and Darwinian (positive) selection [54]. Negative selection operates on genomic regions that must not change in order for an organism to remain viable. The negative selection mechanism is basically death or lower reproductive fitness so that changes to functional regions are not incorporated into later lineages. For example, imagine a gene for 'has blood' in mammalian species. An extreme mutation disrupting the function of 'has blood' in a mammalian offspring would probably mean that it would not survive.

Positive selection applies to regions that might change and where these changes might actually confer a fitness advantage (i.e. adaptation). Changes to these regions are selected for meaning that these regions experience higher degrees of variation be-

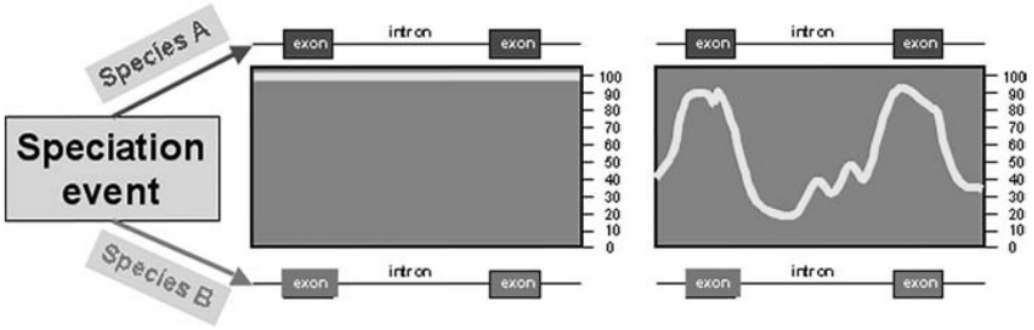

Figure 2.3: Conserved and variable regions, from [54]

cause they do not necessarily cause death. The positive selection mechanism reflects potential adaptation where a change might allow for better survival of a given lineage. Resistance to antibiotics in some bacteria is an example of a positively selected mutation.

Figure 2.3 provides an illustration of these concepts [54]. Two genomic sequences are depicted after a speciation event. A speciation event describes the point at which a genomic sequence has changed so much so as to now define a new species from its ancestors. The exons in Figure 2.3 represent functional regions that are under negative selection. The introns represent variable regions under positive selection. The y-axis measures the amount of base pair variation found between species A and B with respect to these regions. This image illustrates the conservation of regions under negative selection and the variability of regions under positive selection.

When an organism's genome is sequenced, this means that the entire genome is available in string form. Comparing genome strings from multiple organisms allows the determination of both conserved and variable regions. Identifying conserved regions among a group of sequences can aid in the determination of functional regions [50]. Genes associated with virulence in human pathogens have been identified in this manner [78]. Human pathogens are microbial organisms that can cause illness in humans, whereas virulence is the damage caused to the human while infected. Through genome comparisons, the researchers in [78] found 1,024 candidate genes

that were conserved in 90% of the genomes of a diverse range of virulent pathogens such as Anthrax, Botulism, Plague, Smallpox and Tularemia, but were absent from all non-pathogens examined.

Comparing variable regions might indicate adaptive sequence changes in response to such variables as the environment or in the case of pathogens, host immunity [54]. For example, more than 90% of *Influenza A virus* (H3N2) samples collected in the United States in the 2005-2006 flu season were resistance to adamantanes, an early form of antiviral drugs. This represents a sharp increase from the approximately 1%-3% of resistance noted before 2004. The genetic cause of resistance is a single amino acid replacement that is encoded by a three base pair long nucleotide sequence [72].

Comparing genomic sequences can also allow the estimation of evolutionary progression among a group of organisms. This is important because all inferences in comparative biology depend upon accurate estimates of evolutionary relationships [38]. Figure 2.4 represents a phylogeny of the genomes of distantly and closely related genomes.

A phylogenetic tree is a tree graph representing the evolutionary ordering among a group of sequences. The degree of similarity between sequences is represented by their proximity on the tree. Figure 2.4 also depicts terms used to describe the evolutionary breadth of genomic comparisons. Phylogenetic footprinting refers to the comparison of genomes of distantly related species. Phylogenetic shadowing refers to comparisons among closely related species, such as primates and humans. Population shadowing describes comparing genomes among the same species. Some of the limitations associated with phylogenetic trees are discussed in Chapter 3.

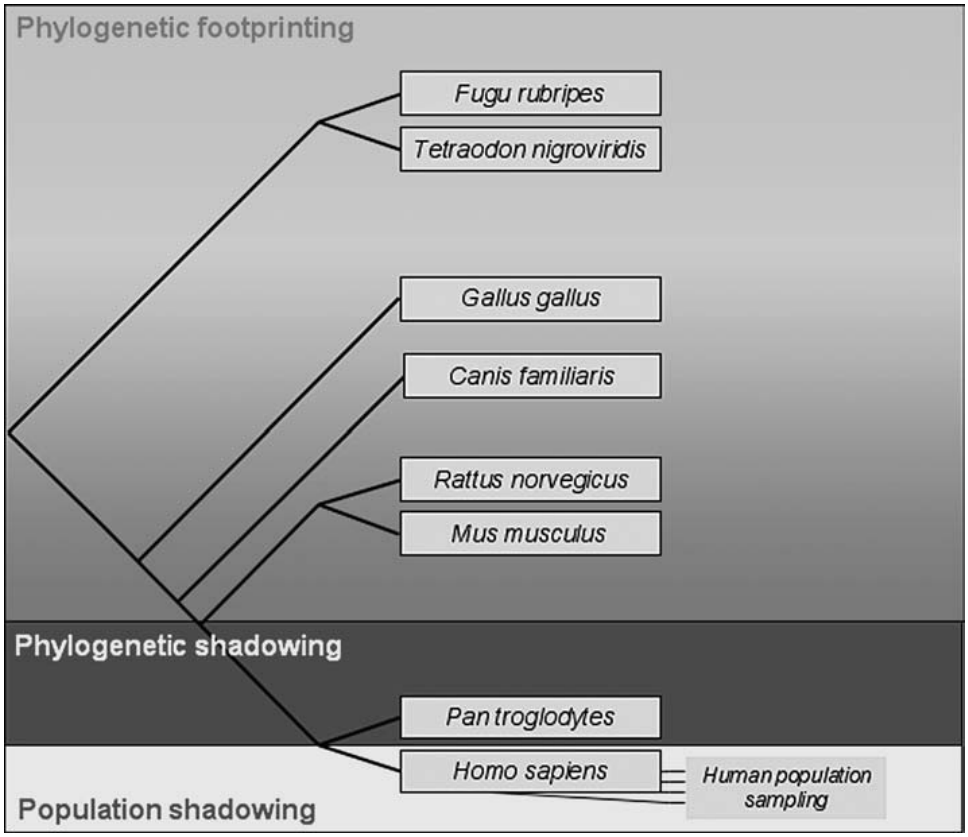

Figure 2.4: Phylogenetic tree, from [54]

Chapter 3

Motivation for Research

3.1 Why differentiate among subspecies viral genomes?

As technology advances, bioinformaticians are incrementally presented with new combinations of increased computing power and larger numbers of complete genomic sequences. Computing hardware has so far adhered to Moore's law, which states that the number of transistors on a circuit board doubles approximately every two years. This has resulted in rapid and consistent improvements in processing speed and memory capacity [76]. At the same time, developing methods in sequencing technology (Next Generation Sequencing) promise to continue to provide faster and cheaper methods for sequencing entire genomes [51, 53].

As a result, large sets of sequences representing bacterial and viral samples within the same species group are publicly available. For example, the *Influenza* A *virus* has been particularly well-sampled. The pandemic potential of Influenza coupled with its high distribution across the globe has resulted in a high rate of sampling, analysis, and surveillance [10]. The National Center for Biotechnology Information (NCBI)'s Influenza Virus Resource database currently houses over 70,000 influenza viral sequences spanning multiple host species, decades, subtypes, and geographic locations [2]. Other publicly available RNA viral databases include the Dengue Virus Resource also hosted by the NCBI Virus Variation Resource [69], and the Los Alamos HIV database [46]. In addition, a West Nile virus database will soon be hosted by the

NCBI Virus Variation Resource. These databases can be expected to continue to grow in the future. Advances in computational methods that combine high-throughput processing capabilities and increasing data set sizes are now the largest challenge to analysis of such viral genomic information [30].

Genomic-based forensics attempts to pinpoint the source of human infections of existing pathogens. Two published studies determined the likely source of human immunodeficiency virus type 1 (HIV-1) through genomic sequence comparisons. The first study published in 2006 investigated the source of HIV type 1 infection of children in a Libyan hospital in 1998 [11]. Foreign medical staff, including a Palestinian doctor and five Bulgarian nurses, were originally accused of infecting 448 children. HIV-1 viruses were sequenced from 51 children and phylogenetic trees of the *gag* gene showed these samples closest to strains already circulating in West Africa. This and further genomic-based analysis led the researchers to conclude that the hospital had had a longstanding HIV-1 infection-control problem preceding the arrival of the foreign medical staff. An earlier study in the United States determined the source of HIV-1 infections for five people with no known risk factors to be a common HIV-1 infected dentist [61]. Figure 3.1 depicts a phylogeny of viral sequences (x and y) collected from the Dentist and Patients A, B, C, E and G.

Somewhat similar to forensics is the approximation of disease transmission networks through genomic sequence comparisons. This genomic approach to epidemiological discovery is becoming increasingly feasible with the current growth in public genomic sequence data.

Current methods for characterizing the spatial and temporal structure of past epidemiological events, which are not based on genomic data, can rely on subjective data collection and/or require extensive research [37]. Due to collection methods of the data required for recreating disease transmission networks, such as patients being required to remember all contacts, resulting graphs can be tree-like [37], relatively small, and misrepresentative of the networks complexity.

Whole-genome comparison methods can provide a quantitative approach to ap-

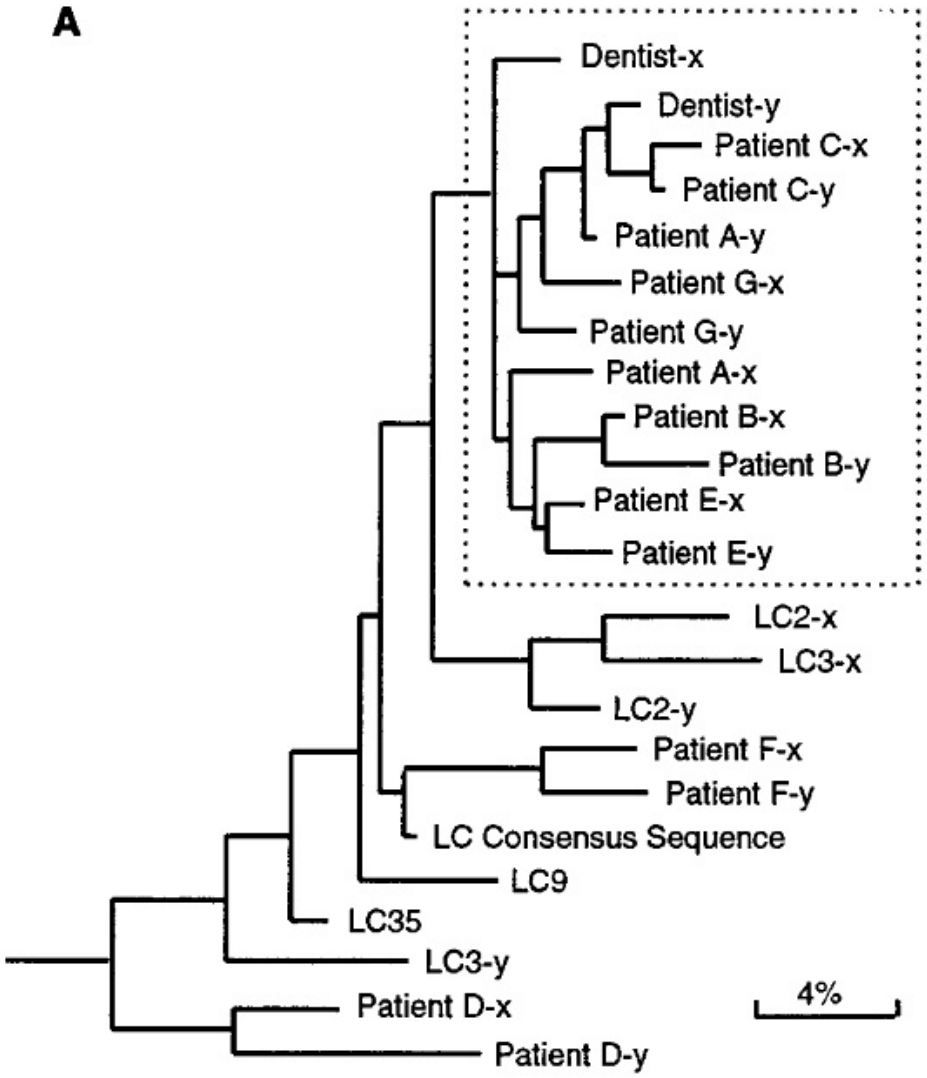

Figure 3.1: Phylogeny of HIV-1 samples (x and y) linking patients A,B,C,E and G to Dentist, from [61]

proximating evolutionary relationships. These methods are particularly attractive in studies of RNA viruses (such as *Influenza A virus*), which mutate rapidly [67] and are relatively small. Small sequence sizes reduce computational requirements. High mutation rates create a potentially traceable micro-evolutionary pathway through sequence comparisons. Many of the viral isolates have known collection locations easily convertible into geographic coordinates. Comparisons of global interspecies isolates with known geographic origins might allow for the identification of global circulation routes and interspecies transmission ('host jumping') patterns, thereby providing mechanisms to develop networks to model disease transmission.

From a graph theoretic approach, a disease network can be viewed as a graph of vertices and edges where individuals or groups of genomic sequences are represented by vertices and their pairwise transmissive potential is indicated by edges. Current methods of developing disease networks from publicly available genomic data are based on phylogenetic tree inferences. These include phylogenies derived from multiple sequence alignments [56] and more recently, Bayesian phylogeography [42]. Phylogenetic trees are basic networks in the form of tree graphs, and are not designed to encompass the amount of data available and required to characterize complex networks. In a phylogenetic tree, each vertex can have only one parent and two children. However, in a complex network, vertices can be connected to a number of other vertices via edges not necessarily based on ancestor/progeny relationships. This approach circumvents phylogenetic tree computation, and provides the consideration of larger datasets and less restrictive edges constraints.

3.2 Comparing whole genomes

Historically, limitations in computing power and sequence data have required researchers to select smaller, orthologous coding regions from groups of genomes for building phylogenies [77]. Orthologous genes are genomic regions that have been conserved among different species after a speciation event, such as the imaginary 'has blood' gene in mammals. Single gene phylogenies were originally used in response to

limits in computing power and data [77]. A common example is phylogenetic study based on ribosomal RNA coding regions. The first reported use of ribosomal RNA was originally based on the following: 1) ribosomal RNA was thought to be found in all self-replicating organisms, 2) it was easy to extract and, 3) it changes very slowly over time allowing that relationships may be detected among distantly related species [87]. Genome comparisons based on ribosomal RNA are still widely used today to derive the prokaryotic branch of the Tree of Life [44] and serve as the gold standard when phenotype data are scarce [75]. Prior to the discovery of ribosomal RNA, building prokaryotic phylogenies had been deemed unsolvable [60].

A problem with using smaller coding regions to represent entire genomes is that the comparisons of different coding regions can lead to different trees [14, 64, 77]. For example, Figure 3.2 illustrates different trees resulting from different genome subsets among species of corona virus genomes [91].

After the year 2000, genome comparisons began using multiple genes and current studies often incorporated more than 100 coding regions [14]. Most likely, these genome subsets prove sufficient for genome sequence comparisons among different mammalian species [14, 31]. However, comparing microbes using only genome subsets is less supported due to the high prevalence of horizontal gene transfer [88] and recombination.

Horizontal gene transfer (HGT) is a major force in archaeal and bacterial evolution [20]. This is a mechanism by which genes from one organism might be incorporated into the genome of another organism that is not an offspring. Thus, an evolutionary tree can be viewed as a more complex network than traditional bifurcating tree representations. This is illustrated in Figure 3.3 from [17]. *Yersinia pestis* [62] and *Neisseria meningitis* [80] are thought to have acquired pathogenicity through this phenomena. While HGT is not assumed to affect ribosomal RNA coding regions [14], it requires consideration when building gene-based phylogenies.

Recombination is another genomic mixing mechanism common in microbial species whereby entire genome segments are inherited for different progenitors [25]. Recombi-

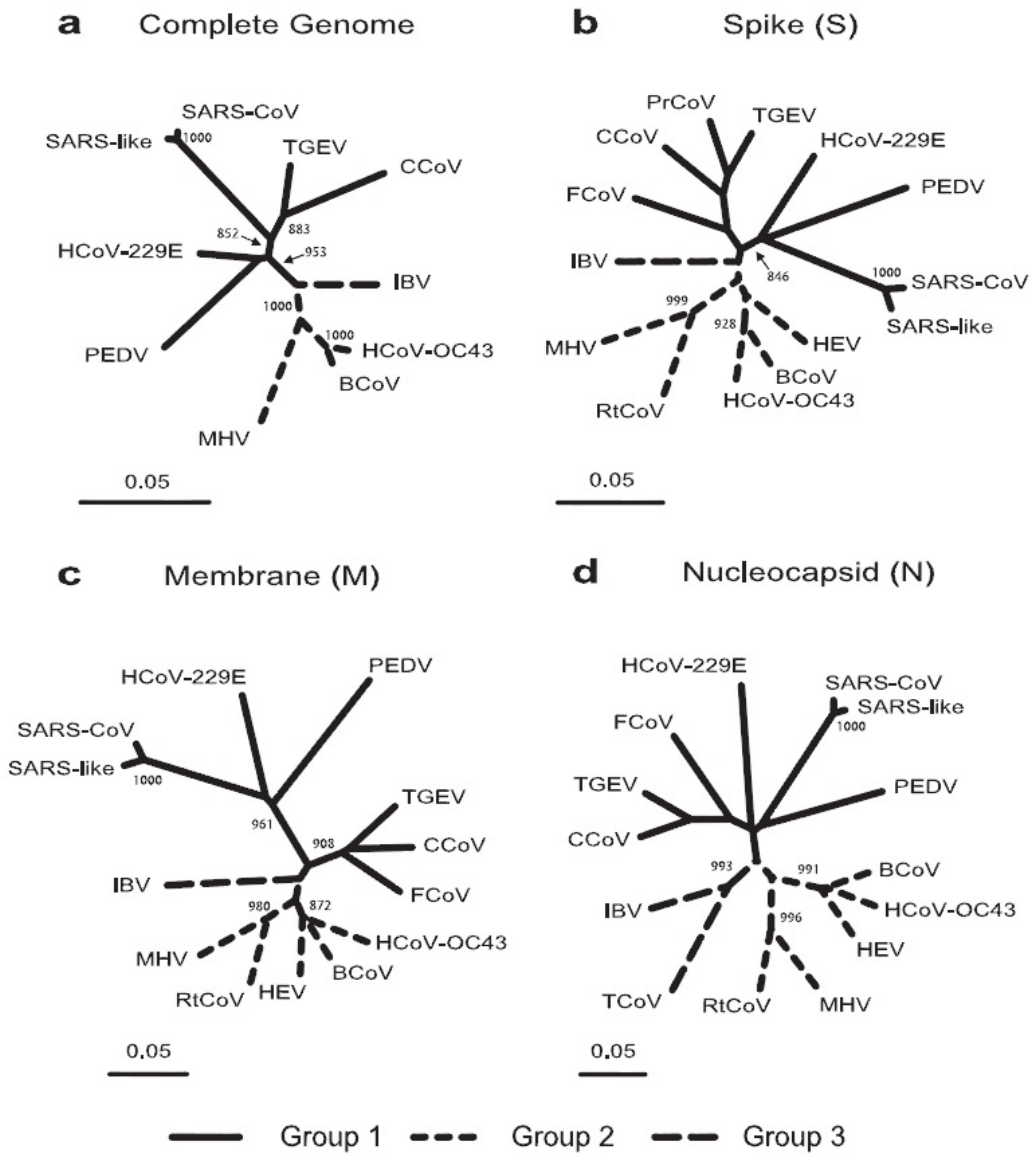

Figure 3.2: Different phylogenies resulting from whole-genome and selected gene comparisons, from [91]

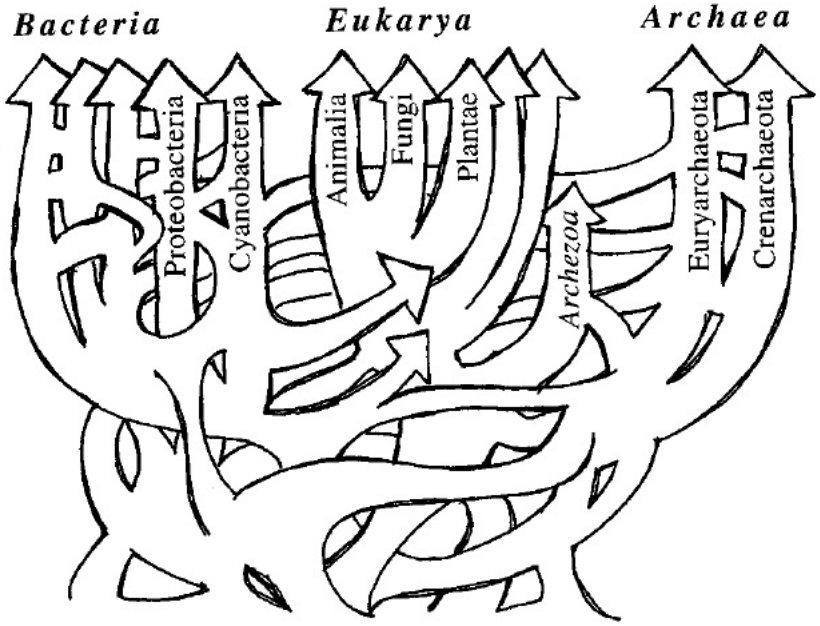

Figure 3.3: Theorized effects of Horizontal Gene Transfer (HGT) on tree of life, from [17]

nation is extremely prevalent in RNA viruses such as HIV and *Influenza A virus* [67]. Figure 3.4 depicts recombination among multiple influenza viruses [30]. In Figure 3.4, two different viruses co-infect the same cell and resulting progeny contain combinations of genome segments from both.

For making lineage distinctions among subspecies microbes it should be optimal to base comparisons on complete genomic sequences. While microbes are rapidly evolving, organisms from the same species will be expected to still have many conserved, coding regions. For example, ribosomal RNA is generally too conserved to distinguish among bacterial strains within the same species [45]. To differentiate among subspecies samples, the inclusion of variable, non-coding regions is useful. Subspecies variation is often referred to as phylogenetic noise [54]. However, among individual samples of the same species, this phylogenetic 'noise' may be necessary for making distinctions among lineages and depicting intra-species phylogenies. To illustrate this, we ran two word-based comparisons on whole genomes and then again on only coding regions of two strains of *Escherichia coli* (*E.coli* 0157:H7 str. EDL933

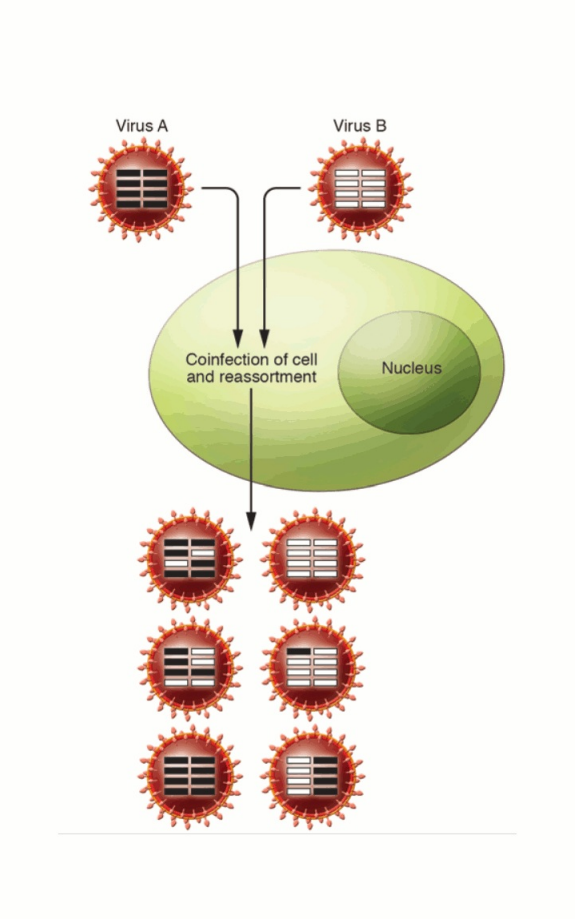

Figure 3.4: Depiction of reassortment between two influenza viruses, from [30]

and *E.coli* 0157:H7 str. EC4115). The coding regions contributed to 14% of the total differences measured between the whole-genomes. Thus, 86% of word-based differences between strains were found within non-coding regions. These percentages merit further investigation into how crucial non-coding differences (phylogenetic noise) are for depicting accurate phylogenetic relationships among closely related strains.

3.3 Computational limits

3.3.1 Multiple Sequence Alignment

Methods commonly used to infer evolutionary trees of genomic sequences are not amenable to large whole-genome sequence datasets. These methods are generally based on pairwise and/or multiple sequence alignments (MSA), originally designed and tested with relatively short protein sequences in mind [9, 19]. Multiple sequence alignment methods have been proven to be NP-complete [21, 83, 84]. In response, heuristics (computational shortcuts) are often employed to reduce computation times. The most common heuristic approach is progressive alignment [19, 59]. In progressive alignment, pairwise alignments are conducted among all pairs of sequences in a group. The overall multiple sequence alignment is then progressively built on these pairwise alignments.

Several commonly used programs based upon sequence alignment methods including ProbCons, T-Coffee, MAFFT, MUSCLE, DIALIGN, ProDA and ClustalW have been reviewed in [19]. In this review, any analysis involving sequences longer than 20,000 characters might cause all program except ClustalW to default on memory. Furthermore, ClustalW would be capable of handling only a small number of these large sequences. While a typical protein sequence will contain approximately 100 characters, complete microbial genomes are generally orders of magnitude larger. Figure 3.5 depicts relative numbers of microbial species and their general size distributions [25]. Figure 3.5 shows that whole genomes of almost none of the microbial species included would fall under the 20,000 character limit suggested in [19].

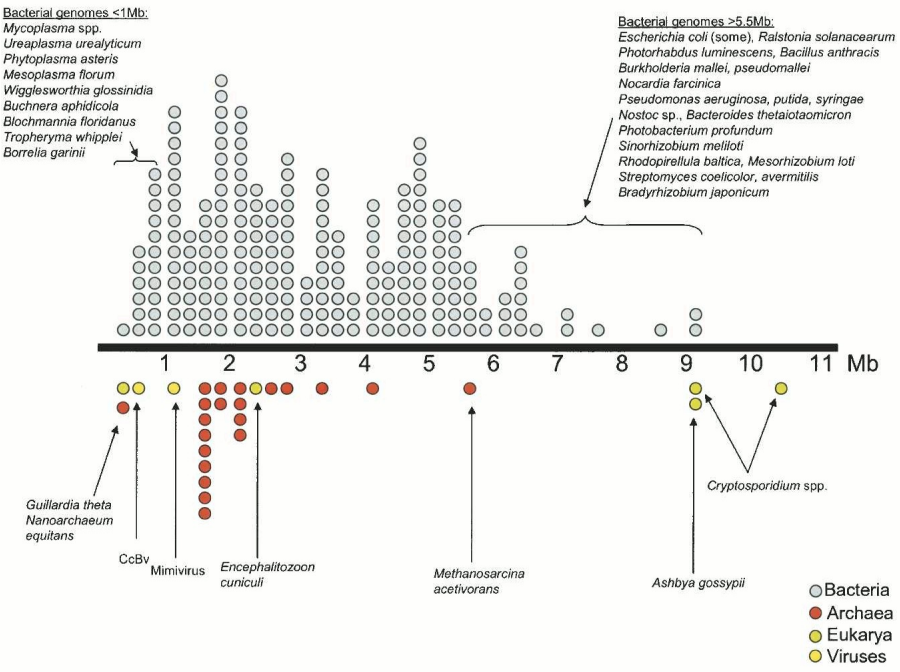

Figure 3.5: Sizes of a set of sequenced microbes, from [25]

As previously mentioned, heuristics enable the solution of an MSA, but they do not guarantee the optimal solution. Thus, different algorithms using different heuristics can yield very different results when applied to the same data set. Uncertainty among MSA algorithms have been described in [47, 49] and illustrated in [89]. MSAs were conducted among seven yeast species based on 1,502 orthologous genes. Seven different applications produced six different resulting trees, Figure 3.6 [89].

3.3.2 Maximum Parsimony

Deriving a phylogenetic tree from a multiple sequence alignment can present a second step which is also NP-Hard [27]. Maximum parsimony is a common approach to tree construction. Given an MSA, the goal is often to construct a tree by finding the arrangement of sequences at nodes that requires the minimum number of changes per nucleotide or amino acid site along each branch. Thus, it is the most parsimonious arrangement of sequences with regards to the number of evolutionary changes

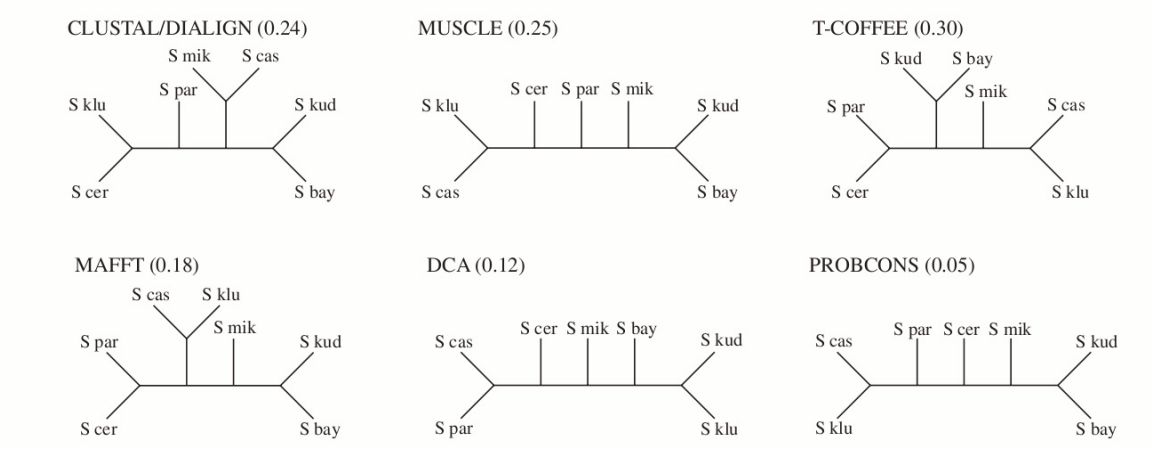

Figure 3.6: Different phylogenies resulting from different MSA programs, from [89]

required to explain the arrangement. To solve this problem optimally, all possible trees constructed from a set of sequences must be tested, and the overall number of site changes tallied for each tree configuration. The tree with the least number of changes, or the least expense if different types of changes are given different costs, is then chosen as the optimal solution. This becomes computationally intractable for even a moderate number of sequences. Given N sequences, the total possible number of rooted trees is $\frac{(2N-3)!}{2^{(N-2)}(N-2)!}$ and the number of possible unrooted trees is $\frac{(2N-5)!}{2^{(N-3)}(N-3)!}$. Table 3.1 provides a list of the possible number of each type of tree given N sequences.

Necessarily, heuristics are often employed to avoid constructing all possible trees. These include methods such as branch and bound [5] and hill climbing [26]. However, as mentioned above, while heuristics guarantee a solution, they do not guarantee the optimal solution. Thus, a parsimonious tree constructed with heuristics may or may not be the best possible solution.

3.3.3 Pairwise Sequence Alignment

Phylogenetic trees may also be constructed from all-against-all pairwise alignments, avoiding multiple sequence alignments altogether [8, 68]. However, pairwise alignment also presents a computational bottleneck because of the basic requirements to derive a solution.

Table 3.1: Number of possible rooted and unrooted phylogenetic trees from a MSA

N	rooted trees	unrooted trees
2	1	1
3	1	3
4	3	15
5	15	105
6	105	945
7	954	10,395
8	10,395	135,135
9	135,135	34,459,425
10	34,459,425	2.13E15
15	2.13E15	8.E21

Alignment algorithms such as Needleman-Wunsch [55] and Smith-Waterman [74] are very similar to the longest common subsequence(LCS) problem and are based on dynamic programming. As such, their time and space requirements are asymptotically bounded by $O(N^2)$ in big-O notation, which is a theoretical measure of the computation required for a given algorithm. Here, N is the maximum sequence length of a sequence pair. This upper bound is explained by the dynamic programming approach of solving and storing solutions to sub problems in the process of progressively solving entire problems. When solving an alignment of two strings, S_1 and S_2, an $|S_1| \times |S_2|$ matrix is created, where $|S_i|$ denotes the number of characters in sequence S_i. To derive a solution, each entry in the matrix is examined at least once leading to the $O(N^2)$ computational bound. Figure 3.7 illustrates this in a depiction of the Smith-Waterman algorithm [74].

In Figure 3.7, matrix (H) elements are computed in a row-wise fashion, beginning at the upper left corner element $H_{1,1}$. Each subsequent value computed at matrix element $H_{i,j}$ depends on the match or mismatch of sequence characters a_i and b_j along with values in the matrix previously computed. Specifically, to determine a value at each position H_{ij}, we must examine the following computed values: 1) the upper left diagonal neighbor $H_{i-1,j-1}$, 2.) the entire computed i^{th} row, and 3) the entire computed j^{th} column.

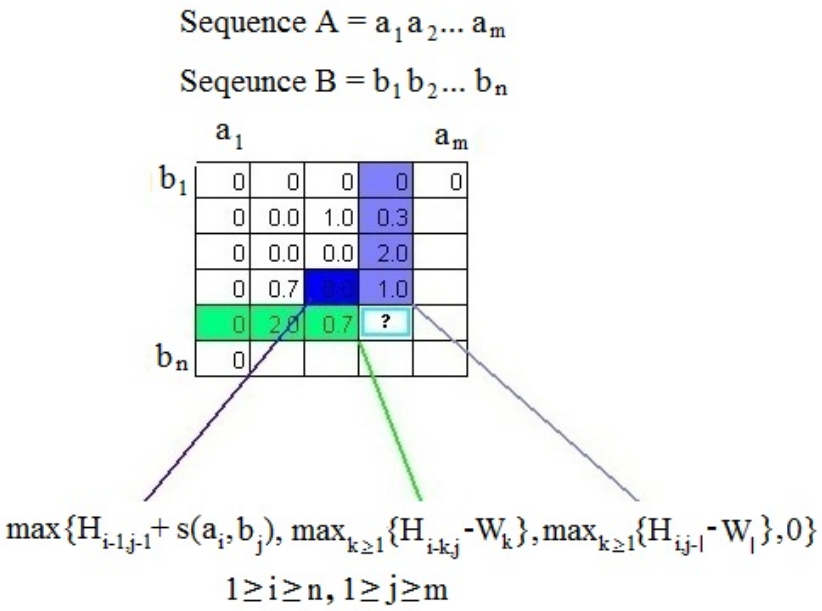

Figure 3.7: Illustration of the Smith-Waterman algorithm

3.4 Alignment-free computation

In contrast to the $O(N^2)$ computational bound for pairwise sequence alignment, alignment-free methods can run on the order of $O(N)$. To compare two sequences, the general approach is to parse through each sequence once while deriving measure (M) of each. Thus, to compare two sequences (S_i and S_j), we need only parse through each sequence once, and then compare values between $M(S_i)$ and $M(S_j)$. Numerous alignment-free methods of genomic comparison have been proposed. In addition to computational advantages, alignment-free comparative approaches are not based on the assumption that matching subsequences across sequence pairs exist in the same order (i.e. conservation of contiguity between homologous sequences) [82]. Thus, one sequence may be 'shuffled around' relative to another without violating this assumption. Because of this, alignment-free measures might have an advantage in comparisons among microbes which are subject to high levels of recombination and lateral

gene transfer. A comprehensive review of alignment-free measures is found in [82].

Chapter 4

k-mer-Based Comparisons of Microbes

4.1 What is a *k*-mer?

Vinga and Almeida present a number of alignment-free sequence comparison measures that are based, in some way, on nucleotide word composition comparisons among sequences [82]. Nucleotide words are often referred to as k-mers, k-tuples or oligonucleotides. A k-mer is a nucleotide word that is k characters long. Due to the small alphabet size of genomic sequences $\{a, c, g, t/u\}$, there exist only 4^k possible nucleotide words for any positive integer value assigned to k. For example, if $k = 2$, then the set of all possible nucleotide 2-mers, or di-nucleotides, $\{aa, ac, ag, at, cc, ..., tt\}$ contains 16 (4^2) words. The same word-based measures can be applied to amino acid sequences, in which case the alphabet size is 20 instead of 4. This research focuses on nucleotide sequences, fixing the size of our alphabet to four letters.

In k-mer-based comparisons, a genomic sequence is parsed only once to determine the number of occurrences (count) of each possible k-mer. Common approaches view each count as a relative frequency, by dividing the count by the total number of k-mers observed in the sequence [73]. Another common method is to divide k-mer observed frequencies by their expected frequencies, which are based on random nucleotide distributions. In this approach, a large ratio indicates that the expression of a specific k-mer is favored by some evolutionary mechanism in a given sequence, an event referred to as usage bias. These observed/expected ratios form the basis of

oligonucleotide profiles, originally presented in [35].

4.2 Computation of *k*-mers

Counting and storing all k-mers present in a given sequence is relatively straightforward and can be accomplished in linear-time using a hash function. A hash function is a mathematical formula used to convert data into a simplified representative numeric value. A simple hash function can be used to convert all possible k-mers for a given value of k into consecutive integers ranging from $0, ..., 4^k - 1$.

For example, each nucleotide is assigned a number (n) such that $n(a) = 0, n(c) = 1, n(g) = 2, n(t) = 3$. Then each nucleotide character in a word is assigned a position ranging from 0 to $k - 1$ increasing in a right to left manner. Thus, given a 3-mer such as act, its character positions are denoted as subscripts $a_2c_1t_0$ and its numeric representation is 013. A single integer value is then computed as:

$$\sum_{p=0}^{p=k-1} n(\text{character at position } p) \times 4^p \tag{4.1}$$

Thus, $act = (0 \times 4^2) + (1 \times 4^1) + (3 \times 4^0) = 7$. The total number of possible 3-mers is $4^3 = 64$ and using the described formula, values for aaa,...,ttt correspondingly range from $\{0, ..., 63\}$.

All k-mer counts in a sequence can be represented by an array of length 4^k. The i^{th} position in the array is associated to the count of the i^{th} word of length k. All 4^k positions are initialized to zero, and a sliding window of length k is used to parse the sequence. Each k-mer detected by the window is converted to its integer value using a hash function as described, and the value at that integer position of the array is incremented. Figure 4.1 depicts a short sequence and its corresponding 2-mer count profile. Lines are drawn between each 2-mer and its position in the array determined by the hash function (Equation 4.1).

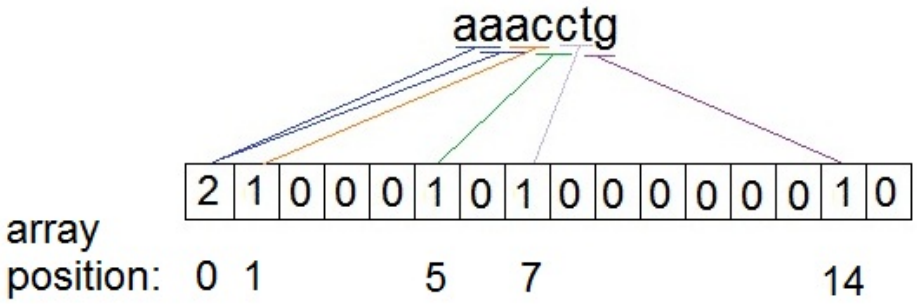

Figure 4.1: Illustration of a 2-mer count array

4.3 Comparing *k*-mer profiles

Methods for comparing two sequences based on k-mer count or frequency profiles are generally performed on a word by word basis. For example, a distance measure between 2-mer profiles of sequences S_1 and S_2 is the sum of differences between counts of all possible 2-mers $\{aa, ..., tt\}$ in each profile. While several algorithms to compare sequences in this manner are similar, they differ in the methods used to compute differences between specific k-mers and whether k-mer values are represented as counts, frequencies, observed/expected ratios, etc. As an example, here we present two word-based comparative methods: the d^2 distance published in 1990 [81] and a method that uses information theory to measure differences among Feature Frequency Profiles published recently in 2008 [90].

The d_k^2 distance between two sequences (S_1, S_2) is formalized by [81] as:

$$d_k^2(S_1, S_2) = \sum_{i=1}^{4^k} p_i \left(c_i\left(S_1\right) - c_i\left(S_2\right)\right)^2 \tag{4.2}$$

where k is the word length, $c_i(S_1)$ and $c_i(S_2)$ indicate counts of k-mer$_i$ in sequences S_1 and S_2, respectively, and p_i is a weight associated with each k-mer. Current uses of this measure are often limited to nucleotide words of length six [29]. However, [81] suggests using a range of word lengths between l and u providing:

$$d^2(S_1, S_2) = \sum_{k=l}^{u} d_k^2(S_1, S_2) \tag{4.3}$$

In a recently published comparative measure based on Feature Frequency Profiles (FFP) in [90], the Jenson-Shannon (JS) Divergence is used to compute distances between k-mer frequencies. The FFP of a sequence first entails a linear-time parsing to obtain counts of each possible k-mer, yielding a count vector C_k. The FFP profile F_k is then obtained by normalizing each vector element in C_k by the total number of k-mers found in a sequence such that, $F_k = C_k / \sum_{w=1}^{4^k} c_{w,k}$.

A measure of dissimilarity between two sequences can then be computed as the sum of element-wise differences between frequency profiles. In order to compute element-wise differences, the Jenson-Shannon (JS) Divergence drawn from information theory is used. Let P_k and Q_k represent FFPs for sequences A and B, respectively, and M_k their average so that $M_k = (P_k + Q_k)/2$. The JS Divergence is then calculated as:

$$JS_k(P_k, Q_k) = (1/2)KL(P_k, M_k) + (1/2)KL(Q_k, M_k) \tag{4.4}$$

where the Kullback-Leibler (KL) Divergence is

$$KL(P_k, M_k) = \sum_{w=1}^{4^k} p_{k,w} \log_2 (p_{k,w}/m_{k,w}) \tag{4.5}$$

Both equations (4.2 and 4.4) described above derive a single distance measure between two sequences based on full k-mer profile counts. The approaches differ in the method of comparing pairwise k-mer count profiles and illustrate the wide range of available k-mer-based comparative methods.

4.4 The case for *k*-mers

4.4.1 Usage bias

Usage bias is a term used to describe the over- or under- abundance of specific nucleotide words in a genomic sequence. It is of interest because differences in bias can

be found between different genomes. Bias is quantified by the ratio of the observed occurrence of existing k-mers in a sequence compared with the expected occurrence in a randomly organized sequence, *a.k.a*, the genomic signature mentioned in Section 4.1. Bias in prokaryotic genomes has been found in k-mers where k ranges from 2 to 6 [3, 4, 34, 35, 36, 65].

The prevalence of bias in prokaryotic genomes suggests that k-mer based measures can address compositional differences at the subspecies level. Biases have been compared among species types (i.e. eukaryotes, prokaryotes, bacteria and viruses), although literature searches indicate that subspecies differences have not been examined in detail.

For example, in [34], di-nucleotide signatures show significantly more difference between species than within species. Bias within species was noted; however, subspecies differences have not become a focus of any further research. Figure 4.2 from [34] illustrates pairwise di-nucleotide signature differences among several microbial species. Identity values show bias differences existing within same species sequences. Findings such as in [34] indicate that k-mer based measures can be useful for comparing subspecies microbial genomic sequences.

4.4.2 *k*-mer presence and absence

Specific nucleotide words have been found to be absent from species of mammals, bacteria, fungi and yeast [28]. Comparing which words are present in one group of sequences and absent from another might yield insight into divergent regions among subspecies microbes. At the microbial subspecies level, researchers in [24] found word absence/presence to show more correlation between genomes within the same species than between genomes of different species. Even so, less correlation was found between same species genomes than was statistically expected, and it was suggested that word absence can offer delineation within species groups as well.

In some of our previous work [6], k-mer difference measures are restricted to only those k-mers exhibiting presence/absence variation among *Influenza A virus*

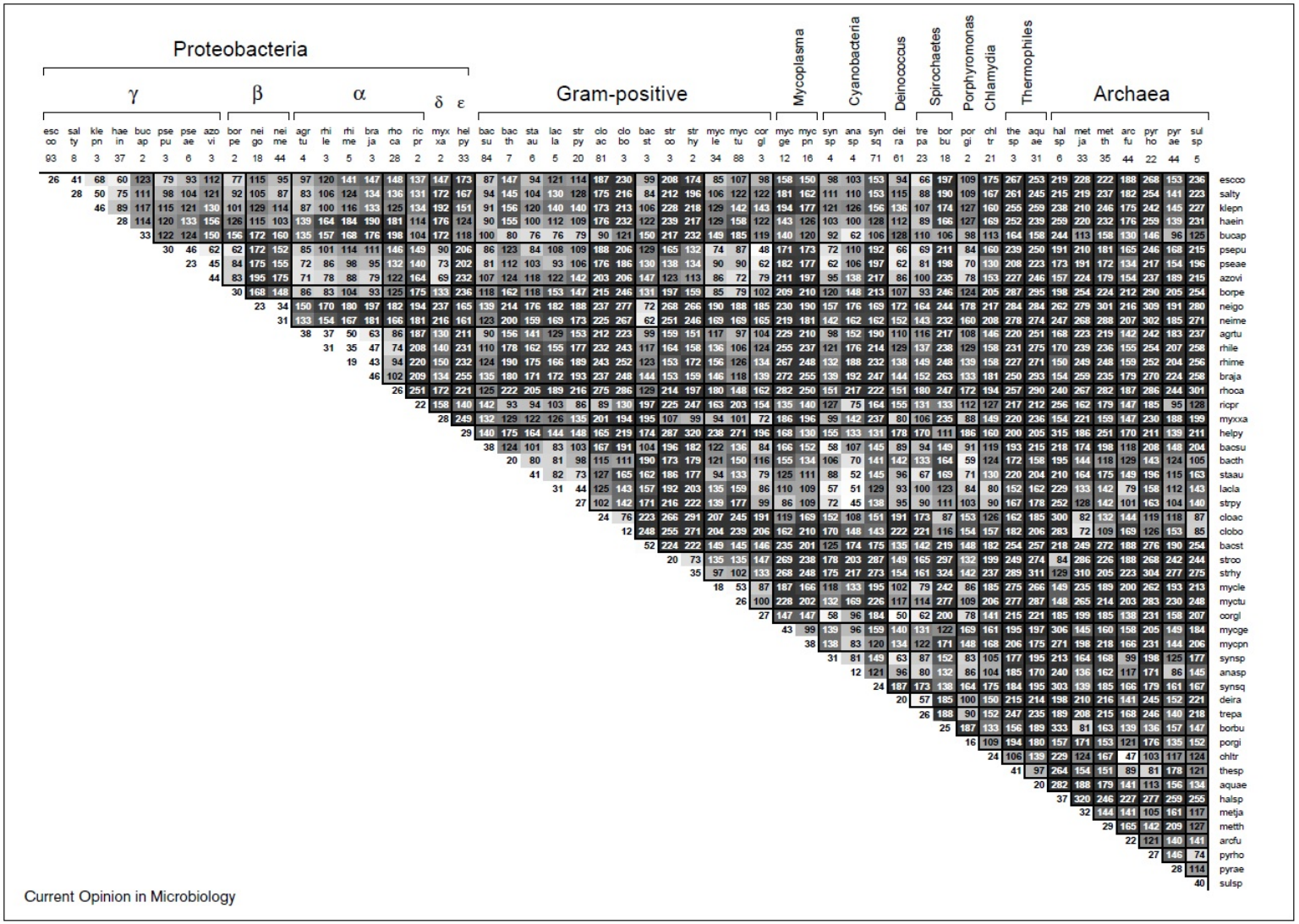

Figure 4.2: Di-nucleotide profile distance matrix, from [34]

whole genomes. Difference measures based on only these subsets enable groupings of whole-genomes by time and location. Calculated sequence differences are shown to be minimal between pairs of sequences originating from the same time and location. Table 4.1 shows a similarity matrix among eight flu isolates from three distinct years and locations. Within each row, the most similar sample based on present and absent k-mers is underlined. In Table 4.1, each sample is most similar to another from the same year and location.

Table 4.1: Inter-epidemic similarity matrix using present/absent k-mers

	Hong Kong 1980		Nicaragua 2007			New South Wales 1990		
	s1	s2	s3	s4	s5	s6	s7	s8
s1	1.000	0.995	0.486	0.480	0.485	0.573	0.573	0.560
s2	0.995	1.000	0.488	0.481	0.487	0.575	0.575	0.561
s3	0.486	0.488	1.000	0.988	0.982	0.656	0.655	0.662
s4	0.480	0.481	0.988	1.000	0.972	0.650	0.649	0.656
s5	0.485	0.487	0.982	0.972	1.000	0.652	0.651	0.658
s6	0.573	0.575	0.656	0.650	0.652	1.000	0.999	0.869
s7	0.573	0.575	0.655	0.649	0.651	0.999	1.000	0.870
s8	0.560	0.561	0.662	0.656	0.658	0.869	0.870	1.000

4.4.3 Markov models

Markov models are commonly used in bioinformatics to compare k-mer compositions between sequences. We have included a description here, although we are not including Markov models in our analyses presented in Chapters 7 and 8 which rely on k-mer counts and frequencies.

In genomic word analysis, Markov models are often used as a means of calculating the expected count of each word (E(w)) in a signature set [43, 65, 70]. In Markov chains, the current state of a system is predicted by its previous states. In word signature analysis, this translates to predicting a word frequency based on the observed frequencies of its subwords or nucleotide content. As described in [65], the ratio of the observed count over its expected count, O(w)/E(w) can then be used to derive the degree of over- or under- representation of each word in found in a given

sequence. Depending on the order of the Markov model, bias contributed to a word of length k from subwords of length *1* to *k-1* can be removed.

With the ultimate goal to match DNA's internal word selection mechanisms, the optimal order of the Markov model to use remains undetermined. Consistent with findings in [65] and [52], minimal order Markov models (Equation 4.6) allowed the most differentiation among genomic signatures of different prokaryotic species.

A minimal order Markov model does not remove subword bias and the expected count of a word E(w), in a genomic sequence of length N is derived as:

$$E(w) = \left[\left(a^{c(w_a)} \times c^{c(w_c)} \times g^{c(w_g)} \times t^{c(w_t)}\right) \times N\right] \tag{4.6}$$

a, c, t and g represent specific nucleotide frequencies in the total sequence S and $c(w_a), c(w_c), c(w_g), c(w_t)$ are the count of each nucleotide in a word w.

4.5 Existing *k*-mer-based comparisons of microbes

Several k-mer-based comparative methods have been used to study nucleotide and amino acid sequences. Methods specifically applied to distinguishing among microbial genomes are based upon all possible k-mers for a single value of k, and not upon specified k-mer subsets. In [91] a method combining k-mer statistics and information theory is described and used to compare subspecies human *Influenza A virus* sequences and for examining the relationship of severe acute respiratory syndrome (SARS) among other corona viruses. Comparisons are performed by ranking 4-mers by their observed counts. A similarity index between two sequences (S_1 and S_2) using k length words is given as:

$$D_k\left(S_1, S_2\right) = \frac{1}{4^k - 1} \sum_{i=1}^{4^k} \left|R_1\left(w_i\right) - R_2\left(w_i\right)\right| \frac{H_1\left(w_i\right) + H_2\left(w_i\right)}{\sum_{i=1}^{4^k}\left[H_1\left(w_i\right) + H_2\left(w_i\right)\right]} \tag{4.7}$$

where $R_1\left(w_i\right)$ and $R_2\left(w_i\right)$ represent ranks of individual k-mers (w_i) in sequences S_1 and S_2, and H is a measure of Shannon's entropy for w_i in sequences S_1 and S_2.

Resulting neighbor-joining phylogenies for both influenza and the corona virus data sets were congruent with current knowledge of evolutionary histories of these

viruses. Figure 4.3 shows an influenza phylogeny derived from the comparative method in [91].

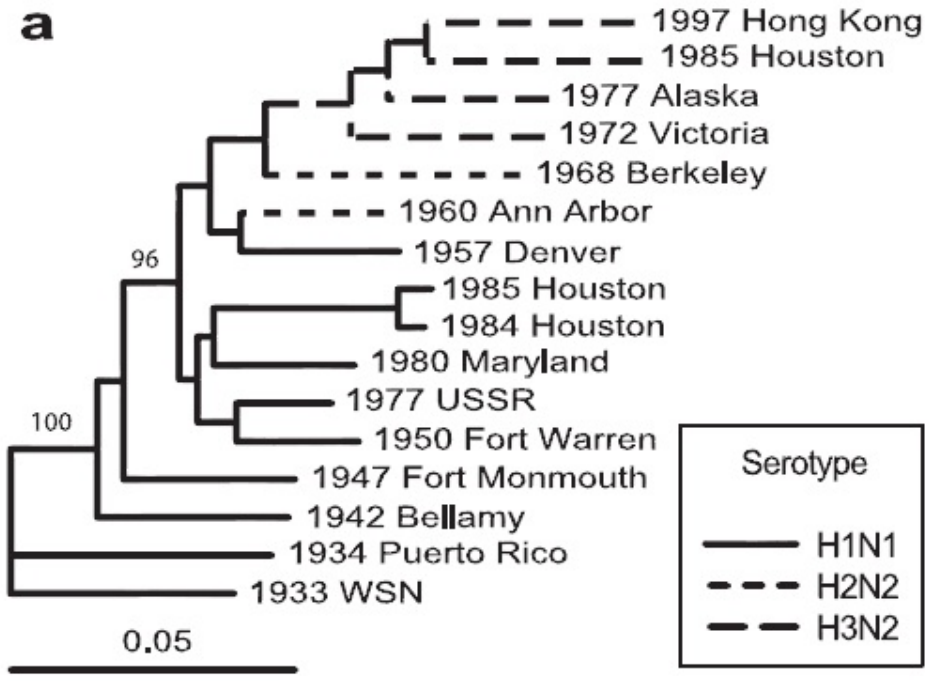

Figure 4.3: Influenza phylogeny, from [91]

Other than analysis of the influenza data set in the previous study (Figure 4.3), current k-mer-based analysis methods generally do not extend further down phylogenetic trees than class distinctions. For example, in [33], the FFP method is used to classify the prokaryotic branch of the Tree of Life down to the class level (Figure 4.4).

In [33], nucleotide sequences of coding regions are used in comparisons. An earlier study addresses the entire Tree of Life using k-mer-based comparisons among amino acid sequences in coding regions [66]. In this study, composition vectors of amino acid words of length 6 (alphabet size = 20) in coding regions are computed from genomes representing Eukaryotes, Archaea, and Bacteria, where composition vectors

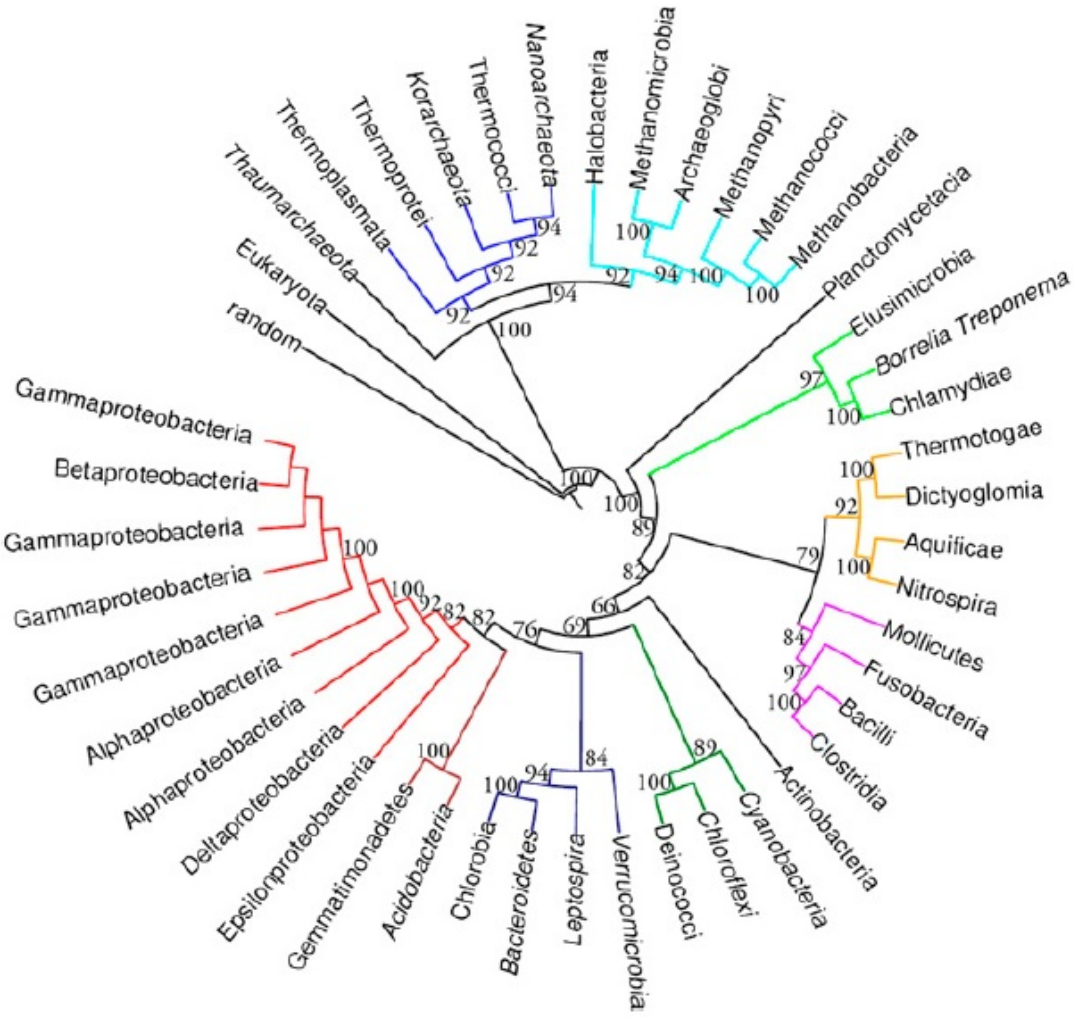

Figure 4.4: Phylogeny of prokaryote classes from [33]

are derived from ratios of actual to expected frequencies of existing words. Correlations among species pairs (X, Y) with observed/expected frequency ratio profiles $X = (x_1, x_2, ..., x_{20^6})$ and $Y = (y_1, y_2, ..., y_{20^6})$ are computed as:

$$C(X, Y) = \frac{\sum_{i=1}^{20^6} x_i \times y_i}{\left(\sum_{i=1}^{20^6} x_i^2 \times \sum_{i=1}^{20^6} y_i^2\right)} \tag{4.8}$$

Results from [66] are shown in Figure 4.5.

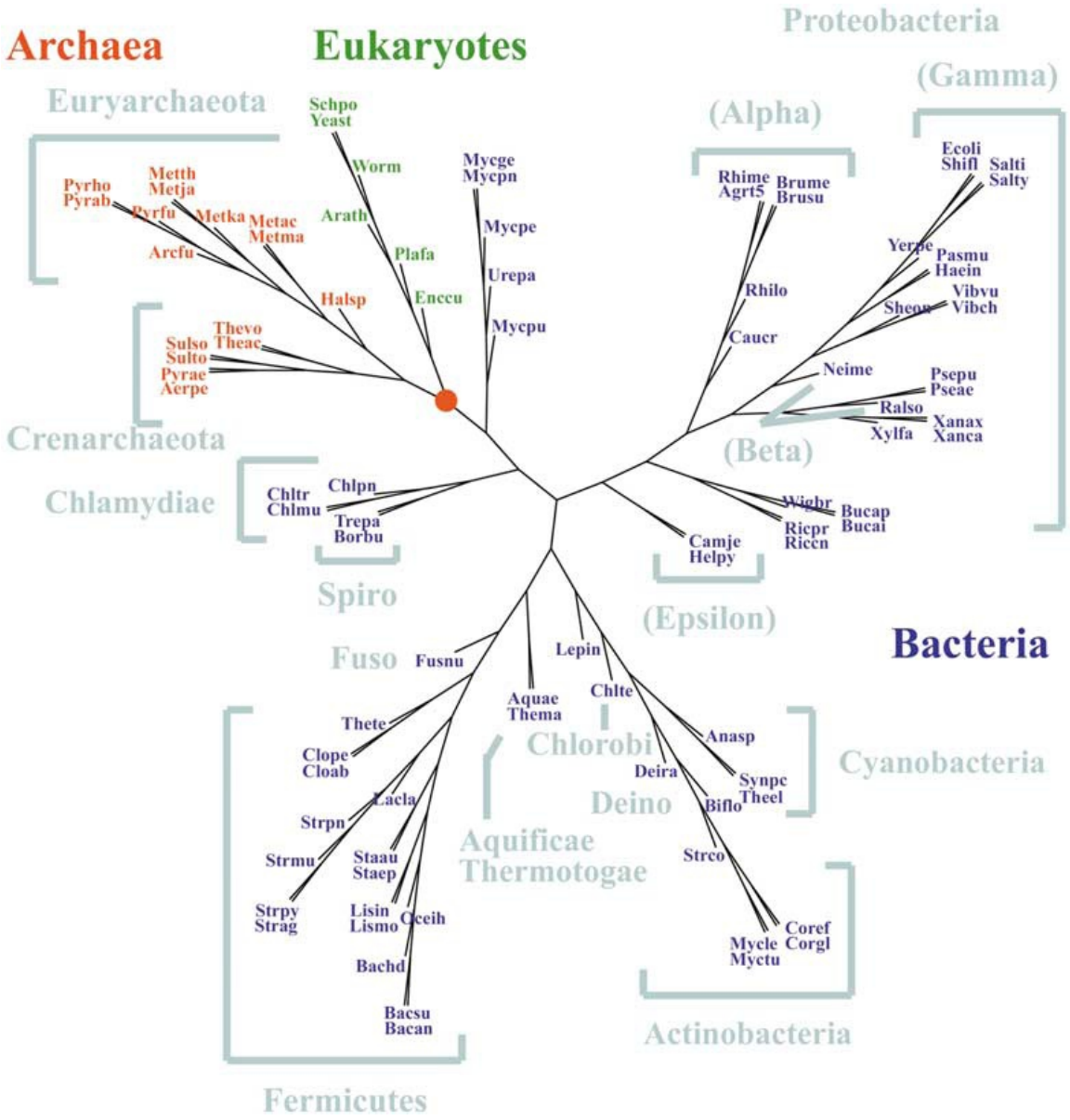

Figure 4.5: Phylogeny of Tree of Life, from[66]

Chapter 5

Proposed Research

5.1 *k*-mer-based subspecies comparisons

The number of publicly available closely related genomic sequences is rapidly increasing, providing a platform upon which to develop computational methods for detecting fine differences between similar whole-genome sequences.

The first stage of our research addresses the computational limitations associated with whole-genome sequence comparisons of large subspecies pathogenic datasets. We then investigate existing methods and develop new algorithms to efficiently and accurately compare whole-genome sequences without alignment. All algorithms are tested on viral intraspecies whole-genome sequences. The ultimate goal of this research is to develop an optimal alignment-free sequence comparison algorithm which approximates existing alignment-based sequence comparison methods. Accurate alignment-free comparisons can enable the computation of pairwise distances among large numbers of very long genomes, and might enable an ordering of their phylogenetic relatedness.

We measure accuracy of all sequence comparison methods by comparing results with those of a standard multiple sequence alignment algorithm (ClustalW). Optimally, our proposed method will serve as a reliable alignment-free replacement to alignment-based comparisons, thereby enabling very quick analysis of large whole-genome sequence datasets. Our work draws from word-based research involving microbial sequences in published literature such as in [34].

5.2 Delineating complex *Influenza A virus* networks

A recent article presenting an influenza study states that there has been "no rigorous measurement of viral diversity across time, across space, and among subtypes" [67] . We try to address this void in the second stage of our research. Here, we propose a graph theoretic approach to investigate disease networks. Current methods of using genomic data to develop disease networks have emerged in response to the increase in data availability; however, many current methods are based upon phylogenetic tree inferences [42, 56], which are not designed to encompass large amounts of sequence data. phylogenetic trees are also limited in their ability to delineate detailed transmission patterns [30, 56, 67]. In contrast, graph-based approaches might bypass many of the computational and structural restrictions associated with phylogenetic trees and might be used to investigate very large and comprehensive data sets.

Our goal is to create disease network graphs from large sets of genomic sequences in which each viral sequence is viewed as a vertex, and edges are drawn between nodes to represent high degrees of sequence similarity. When each sequence is associated with a geographic location and collection date, this representation might provide an approximation of the transmission route of a disease through a series of geographically distributed hosts. The underlying assumption is that a strong degree of sequence similarity indicates the best estimate of transmission given the available data. This approach circumvents the necessity of phylogenetic tree computation, which is computationally intensive and restrictive in the number of edges which can be placed between vertices. To incorporate large sequence datasets, we utilize computationally efficient comparison algorithms examined or developed in the first stage of the proposed research.

5.3 Outline of proposed work

This section provides an outline of the proposed research described in the previous two sections.

Stage 1: Development - develop a computationally efficient k-mer-based alignment-free comparative algorithm and assess the accuracy of the metric.

- **DATA**: Collate several publicly available subspecies viral sequence data sets. Annotate sequences with respect to user-specified characteristics.
- **TESTING**: Develop and test results from different k-mer sequence comparative algorithms through comparisons with ClustalW alignment scores. Select the most accurate algorithm.

Stage 2: Application - using the best algorithm from Stage 1, we will incorporate this into a methodological approach for building graphs from subspecies genomic sequences.

- **DATA**: Collate a selected subspecies genomic data set.
- **COMPARISON**: Derive all pairwise similarity scores using the algorithm selected from Stage 1.
- **GRAPH**: Derive vertex and edge graph where vertices represent sequences and edges are based on similarity scores and thresholds.
  - determine a threshold that approximates a desired degree of sequence similarity for edge placement
  - the accuracy of the threshold will be tested using selectivity and sensitivity measures
- **GRAPH ANALYSIS**: Examine the graph structure and characteristics using existing tools and methods.

  - examine the similarity among groups and across groups of sequence groups via a concept called 'mixing patterns'

- **Example illustrating these methods using *Influenza A virus***: derive an *Influenza A virus* transmission network model from several thousands of whole-genome sequences using the described methodology.

Future Work:

- Address skewed distribution of viral samples to enable more theoretical graph analysis (i.e., network structure).

- Further development of an alignment-free metric to approximate dynamic programming alignment scores.

- Generate graphs based on directed, rather than undirected, edges representing transmission from isolates collected at an earlier date to those collected at a later date.

- Study of more complex connectivity properties of graphs such as connected components and cliques.

Chapter 6

An Annotated *k*-deep Prefix Tree

The following section provides a short summary of an algorithm that we term an "annotated k-deep prefix tree". We use this data structure to conduct a number of word-based comparisons among multiple sequences in a computationally efficient manner. These comparisons are described in Chapter 7. The data structure stores all *1-k*-mer counts for groups of sequences, where *1-k*-mers describe nucleotide words ranging in length from *1-k*. Word-based comparative algorithms can then be conducted on counts stored in the tree without requiring its reconstruction.

The annotated k-deep prefix tree is based upon prefix trees and tree node annotation, both of which are tools often used for string processing in computer science. Tree-based structures are also used in bioinformatics to compare substrings among sequences. The k-deep prefix tree is constructed from the k first characters of all non-empty prefixes of a single or set of genomic sequences. The number of nodes required is exponentially proportional to tree height (k) rather than the total length of sequences, as is the case with suffix trees. A comparable structure was used in [3] to compare the 12-mer 'languages' of human chromosomes 21 and 22. Prefix-trees are also used in the assembly program SSAKE (Short Sequence Assembly by progressive K-mer search and 3' read Extension)[85] to locate overlapping 25-mers between short nucleotide fragments.

The described algorithm provides a compressed and partially or fully dynamically allocated index to all substrings up to a given length (k) found in a single or groups of sequences. While index-based hash tables are generally used for this same purpose,

prefix trees can be more comprehensive, as they might include information regarding nucleotide words of multiple lengths. Hash tables generally represent nucleotide words of a single length, and require large, contiguous blocks of memory for fast look-up times. A prefix tree can be implemented with dynamic memory and is equivalent to multiple hash tables for each word length *(1,..,k)*, with direct links between each word and its prefix and suffix(es).

The benefit of the algorithm is that *(1-k)*-mer compositions of multiple sequences may be counted, stored, and compared efficiently in both time and space in a single analysis. A k-mer denotes a substring (word) of a genomic nucleotide sequence of length k; a *(1-k)*-mer is a word of length k or less. Specific subsets of *(1-k)*-mers selected on the basis of statistical characteristics or biological attributes (e.g. presence/absence variation or high GC content) might easily be highlighted for comparisons at run time.

The word length k best suited in k-mer-based comparative measures is often arbitrarily chosen and must be addressed as a research question in its own right [73]. A fully annotated k-deep prefix tree allows data exploration and the inclusion of multiple word lengths in a single analysis.

Figures 6.1 (a-f) illustrate the generation of a 3-deep prefix tree from the sequence 'CATGAT'. A root node denoting the empty string is created, and successive nodes are built as sequences are parsed and new words are detected. In Figures 6.1(a-f), a single sequence is parsed by a single-spaced sliding window of length k, where $k = 3$. Each nucleotide string determined by the window is inserted into the tree. Each tree node can point to up to four children 'A'-child, 'C'-child, 'G'-child, 'T'-child. The default value for all of a node's children is set to NULL. If a word path in the tree does not yet exist, then it is built upon insertion. In this way, no memory is wasted on nodes that represent non-existent words in the set of sequences being examined.

After the tree has been built, each node represents the termination of a substring found in the sequences(s) parsed. Determining the nucleotide word ending at any node requires a trace-back through parent pointers from that node to the root (Figure 6.2).

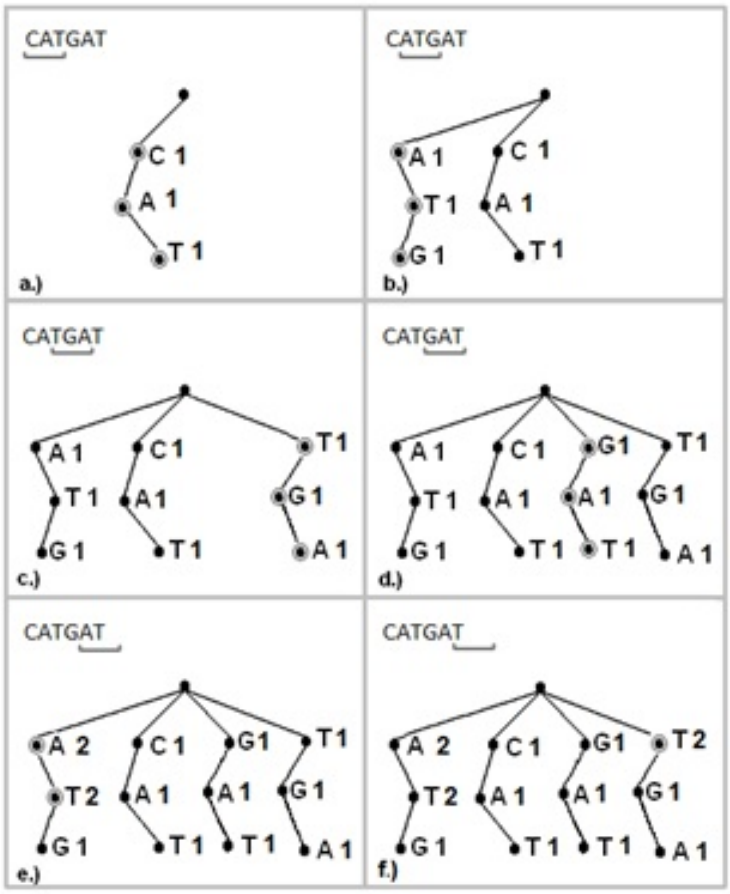

Figure 6.1: Building a 3-deep prefix tree [7]

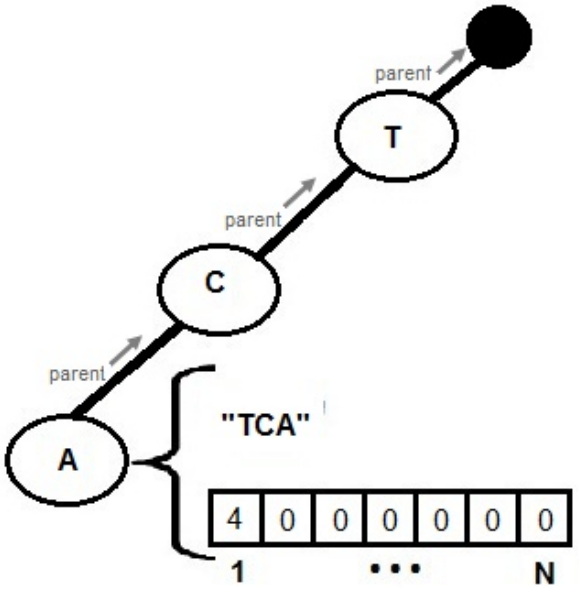

Figure 6.2: Tracing from a node back to the root [7]

This is accomplished in time linear to the length of the word.

Each node has associated with it a count list to enumerate the number of times a word exists within a sequence. A count list is of length N, where N is the number of sequences being compared. When parsing sequence S_i, only position i in the count list will be incremented to count the number of times that word exists (so far) in sequence S_i. Figure 6.3 illustrates a 3-deep tree built from the sequences 'CATGA' and 'ATCAT'. In Figure 6.3, *(1-k)*-mer counts for 'CATGA' are stored in the first index of the count list at each node. Counts for 'ATCAT' are stored in the second index. By maintaining node based count lists, a single tree can contain the complete *(1-k)*-mer composition of multiple sequences. This can then lead to all-against-all k-mer-based sequence comparisons in a single tree traversal.

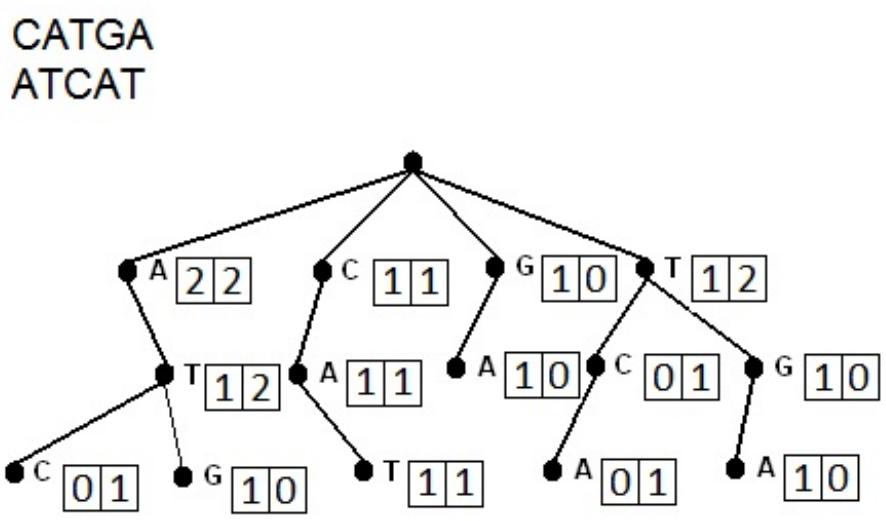

Figure 6.3: A 3-deep prefix tree built on two sequences [7]

Chapter 7

Evaluating Distance Metrics Based on an Accuracy Measure

7.1 Overview

To select a meaningful k-mer-based method for comparing whole genomes, we assess the performance of several k-mer distance metrics by comparing each with ClustalW [40] alignment scores. ClustalW pairwise alignment scores are computed as the percentage of identical bases between two sequences that have been aligned via dynamic programming as described in Section 3.3 and thus provide optimal pairwise alignment scores. Other studies have also used ClustalW alignment scores as a reference for testing k-mer-based methods on a amino acid sequences [18] and short nucleotide fragments [79].

We first convert the ClustalW alignment scores into dissimilarity or distance measures by simple conversion. We then use these ClustalW alignment scores as reference scores to compare k-mer-based alignment-free distance metrics via a simple correlation computation. Higher correlation between the ClustalW reference scores and the k-mer-based distance scores indicates a better performance (greater accuracy) in the k-mer approach. The goal is to determine which rapid k-mer based method provides the best approximation of slower and more computationally expensive full alignments as provided by ClustalW.

The k-deep prefix tree, described in Chapter 6, is developed to store counts of all *(1-k)*-mers in sequence sets, and all k-mer count algorithms are easily and

efficiently implemented using this annotated tree data structure. Using three whole-genome viral datasets for testing (described in Sections 7.11.1, 7.11.2, and 7.11.3), we select five subsets of 20 sequences from each. We then compute pairwise ClustalW reference scores, k-mer-based distances based upon existing k-mer metrics and some with our own modifications. Existing methods include the d_k^2 metric and Feature Frequency Profiles (FFP) methods described in Section 4.3 and the Edgar k-mer distance from [18].

The k-mer metrics we consider here offer a range of diversity. The d_k^2 metric is a sum of squares difference of all existing k-mer counts between two sequences. The FFP method uses information theory by computing the Jensen-Shannon Divergence of individual k-mer frequencies between two sequences. Finally, the Edgar k-mer distance from [18] is based on the maximum number of k-mers co-occurring in the two sequences being compared. This measure is of particular interest, as it includes sequence length in its computation of distance, which likely is an important factor when examining the difference in word composition between two sequences of different lengths.

We make slight modifications to the d_k^2 metric and the Edgar k-mer distance metrics and assess the accuracy of each method with and without modification. Each method and all modifications are described in detail in the following sections.

7.2 Data sets

The data sets used here are composed of subspecies genomes of the single-stranded RNA viruses *Influenza* A *virus*, Human Immunodeficiency virus (HIV), and Dengue virus (DENV). Five sets of 20 sequences are randomly selected for comparisons from each data set: DENV, HIV, and each of the eight influenza segments (*InfA1*, *InfA2*, *InfA3*, *InfA4*, *InfA5*, *InfA6*, *InfA7*, *InfA8*). Data sets are described in detail in Section 7.11.

We consider here RNA virus genomes, as they are relatively small. Small sequence sizes reduce computation requirements and allow us to compute the reference

alignment scores using ClustalW. The maximum input file size that ClustalW allows is 10MB; a file of this size can include only two *E.coli* sequences or 966 DENV sequences. Table 7.1 shows the average sequence length for each data set used in our comparisons, with DENV sequences having the longest average sequence lengths of 10,644 nucleotides. These viral species contain single stranded RNA genomes, so considering the complementary strand of sequences was not necessary.

Table 7.1: Average sequence lengths

Sequence Type	Average Length
InfA, PB2	2,304
InfA, PB1	2,306
InfA, PA	2,188
InfA, HA	1,725
InfA, NP	1,530
InfA, NA	1,422
InfA, M1/M2	992
InfA, NS1/NS2	853
HIV	9,043
DENV	10,644

7.3 ClustalW

ClustalW [40] is a commonly used program that can be used to compute sequence alignment scores using dynamic programming. Dynamic programming results in the most optimal alignment between two sequences, but is computationally expensive. We use ClustalW's "full alignment" option to compute alignment scores based on dynamic programming. Using this option is described in its documentation as slow, but accurate.

For all sequence datasets, ClustalW pairwise alignment scores are computed using the full alignment option and all other settings left at the default values. The default gap opening penalty is 10, the gap extension penalty is 0.1 and the DNA weight matrix is IUB (this contains scores and costs for matching and mismatching nucleotides). A gap opening penalty is the penalty incurred by introducing a gap in an alignment. The

gap extension penalty is the penalty for extending this gap per nucleotide. The weight matrix provides the penalties for mismatched nucleotides. All of these penalties are used to determine the 'best alignment', referred to in Eq. 7.1.

Resulting scores are integers which range between 0-100, where 100 indicates a near perfect alignment between two sequences.

Alignment scores using ClustalW are computed as:

$$\frac{\#\text{identities in best alignment}}{\#\text{nucleotides compared (gaps excluded)}} \tag{7.1}$$

It should be noted that the ClustalW's "best alignment" score is dependent on the gap opening and gap extension penalties. If these penalties are high, sparsely aligned characters (characters separated by large gaps) will be omitted as the gaps required to include them are too expensive in the best alignment calculation. Decreasing these penalties will promote the inclusion of more sparsely aligned characters and can yield higher pairwise alignment scores in certain cases.

7.4 Accuracy assessment

All pairwise alignment scores among the 20 randomly selected sequences are computed using ClustalW and all pairwise distance scores are computed using the various distance methods. Both sets of scores are ordered identically and the Pearson correlation coefficient is computed between sets to measure how well various distance measures approximate full alignment. This process of randomly selecting 20 sequences, computing pairwise alignment and distance measures, and computing correlation coefficients is repeated over five iterations. Correlation scores per method and dataset are averaged over the five iterations.

Because distance methods generate distance scores and ClustalW generates similarity scores, ClustalW similarity scores (percentages) are converted first to distances to coincide with distances generated by the metrics tested by subtracting scores from 100. Each converted ClustalW score is then associated to its corresponding pairwise k-mer distance score so that each set of scores between every sequence pair is

represented by <100 - ClustalW score,k-mer distance score>.

The Pearson correlation coefficient (r) between each set of scores (X, Y) containing n values is computed as:

$$r_{xy} = \frac{\sum_{i=1}^{n} (x_i - \overline{x})(y_i - \overline{y})}{(n-1)\, s_x s_y} \tag{7.2}$$

where $\overline{x}$ and $\overline{y}$ are mean scores, and s_x and s_y are standard deviations of score sets X and Y. Standard deviation is calculated as:

$$s_N = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (x_i - \overline{x})^2} \tag{7.3}$$

7.5 *k*-mer alignment-free distance metrics tested

This section includes a summary of each k-mer based distance method tested. Each method utilizes a single value for k. Testing for each method is conducted using k values ranging from 3 to 13. We find that examining word lengths beyond 13 (up to 20) on several datasets is computationally challenging and does not improve results.

7.5.1 d_k^2

The d_k^2 sequence comparison method is presented in Section 4.3. Briefly, it is the sum of weighted squared differences between counts of all existing k-mers found in two sequences (A, B). It is given as:

$$d_k^2(A, B) = \sum_{i=1}^{4^k} p_i \left(c_i(A) - c_i(B)\right)^2 \tag{7.4}$$

The first version of this distance method tested sets all weights (p_i Eq. 7.4) equal to 1. We refer to this method as **D2** in the following discussions.

7.5.2 Presence/Absence weighting

The second version of **D2** method employs a presence/absence weighting scheme. In previous work [6], we compare influenza genomic sequences using observed-to-expected ratios of k-mer frequencies [65] and limit our study to only those k-mers

that exhibited variation of presence and/or absence across sequences. These are k-mers present in at least one sequence and absent in at least one other sequence in the data set. We found that this method, when restricted to the subset of k-mers exhibiting presence/absence variation, classified sequences from the same epidemic as most similar.

This approach of selecting the presence/absence subset is our design. It is motivated by the idea that locally misaligned regions between two sequences will result in a higher number of k-mers which are absent from at least one sequence. One mismatching nucleotide between two aligned regions forms a part of k overlapping words, and can potentially introduce k words to one sequence that are not present in the second.

We use a simple weighting scheme to instate the presence/absence method: any k-mer that is absent from at least one sequence and present in at least one sequence is assigned a weight of value 1; all other k-mers are assigned a weight of 0. Then if p_i represents the weight for word w_i and $c_{w_i}(a)$ the count of w_i in sequence $S_a, a \in N$, then p_i in Equation 7.4 is given as:

$$p_i = \begin{cases} 1 & if \ \sum_a^N c_{w_i}(S_a) > 0 \text{ and } \prod_a^N c_{w_i}(S_a) = 0 \\ 0 & else \end{cases} \tag{7.5}$$

We refer to this method as **D2PA** in the following discussions.

7.5.3 FFP

The Feature Frequency Profiles (FFP) sequence comparison method is presented in Section 4.3. For each k-mer found in a pair of sequences, the Jenson-Shannon Divergence between the two frequencies of occurrence of that word is computed and summed. The Jensen-Shannon Divergence stems from information theory and it gives a normalized estimate of how divergent two values are. Please refer to Section 4.3 for more detail. This method is referred to as **FFP** in the following discussions.

7.5.4 Edgar *k*-mer distance

A distance measure referred to as the k-mer distance is presented in [18]. This measure divides the maximum number of co-occurring k-mers in two sequences by the total numbers of k-mers found in the shortest sequence (the total number of possible co-occurring k-mers). In the initial step, the Fractional k-mer count F is computed between two sequences (S_1, S_2) where:

$$F = \sum_{i=1}^{4^k} \frac{min\left[c_{S_1}(i), c_{S_2}(i)\right]}{\left[min\left(|S_1|, |S_2|\right) - k + 1\right]} \tag{7.6}$$

where $c_{S_1}(i)$ and $c_{S_2}(i)$ denote counts of k-mer i in sequences S_1 and S_2, and $|S_1|$, $|S_2|$ denote the lengths of both sequences.

The minimum sequence length must be at least k.

The k-mer distance Y is then derived through the transformation:

$$Y = log(0.1 + F) \tag{7.7}$$

if $F \leq 0.9$, this transformation will result in a negative value. The log transformation was implemented in [18] to approximate a better linear relationship between ClustalW scores and F scores.

This method is referred to as **KMER** in the following discussions.

7.5.5 Modified Edgar *k*-mer distance

While computing the correlation between the Edgar k-mer distance and the ClustalW similarity scores, we find that decreasing the constant 0.1 in Equation 7.7 by one order of magnitude can result in a greater range of distance scores in our specific sequence sets that have a greater range in ClustalW alignment scores. This suggests that it may approximate alignment scores for a wider range of datasets. While we are pleased that this modification resulted in a greater correlation with ClustalW alignment scores, we have not tested its robustness with regard to other datasets and have not compared it to other potentially better transformations.

Our modification is illustrated by comparing ranges in scores for both the original and modified Edgar k-mer distance on highly similar and more divergent datasets. The *InfA4* (HA) data set shows a greater range in alignment scores than the *InfA8* (NS1/NS2) data set. Using the original Edgar k-mer distance, the average range in distance scores for the *InfA6* (NA) dataset is 2.32. This range in scores is increased to 4.05 by using the modification we describe here. In contrast, the range in scores remains 1.97 using the original and the modified metric on the *InfA8* (NS1/NS2) segment.

Our modification is noted as:

$$Y = log(0.01 + F) \tag{7.8}$$

if $F \leq 0.99$, this transformation will result in a negative value. We refer to this method in the following discussions as **KMOD**.

7.6 Results

All four distance metrics, excluding the **FFP** method, exhibit high levels of correlation with ClustalW scores. In datasets composed of very similar sequences exhibiting a small range in alignment scores, all four methods are in nearly perfect correlation with the reference ClustalW scores. In contrast, the **KMOD** method shows greatest correlation with ClustalW when using datasets with more divergent sequences that show a greater range in ClustalW alignment scores.

Results are discussed in more detail for each dataset in Section 7.12. A summary of the ClustalW alignment scores computed for all sequence subsets is also included with results. This shows the varied ranges in alignment scores for different datasets. Tables are included in the results, which display the maximum, minimum, range, average and standard deviation of ClustalW alignment scores. These values are averaged over the five randomly selected sets of 20 sequences per sequence set.

7.7 Summary and conclusions

Results here suggest that the best alignment-free k-mer method to approximate alignment is influenced by the range in pairwise similarity scores for a set of sequences. **KMOD** achieves the highest correlation scores for sequence sets exhibiting wider ranges in ClustalW alignment scores; *InfA* (HA), *InfA* (NA), *InfA* (NS1/NS2), and HIV. All methods excluding **FFP** achieve perfect or near perfect correlation with ClustalW scores for all other *InfA* segments with lower sequence divergence. The DENV dataset is also characterized by a relatively low range in sequence alignment scores; however, only the **KMOD**, **D2**, and **D2PA** distance method result in perfect correlation with the ClustalW reference scores.

In all datasets, the **KMOD** method is either superior or equal to any other method tested, which is likely due to the log transform, as it approximates a more direct linear relationship with ClustalW scores. The constants in Equations 7.7 and 7.8 is necessary because F scores (Equation 7.6) have the potential to be equal to zero. This happens in the case when two sequences have no identical k-mers, an event that becomes more probable with shorter sequences and larger k values. The best value for the constant (0.1, 0.01, etc.) may vary with different datasets.

As sequences within the selected subspecies datasets are very similar, overall high levels of correlation with ClustalW alignment scores are not a surprising result: distance measures generated by the Edgar k-mer distance (Equation 7.7) between very similar sequences have also shown a high degree of correlation with amino acid alignment scores generated by ClustalW [18] and short nucleotide sequence alignments with the Needleman-Wunsch algorithm [79]. The constant 0.01 in the **KMOD** log transformation of Equation 7.8 was chosen experimentally. We expect that there might be other transformations that will result in equal or better correlations with ClustalW scores, however, we have not explored this fully. Unsurprisingly, the FFP method shows the lowest levels of correlation with full alignment scores, however, Sims *et.al.* state that accuracy should not be expected between highly similar se-

quences [73].

7.8 Comparisons with MUMmer

MUMmer is a program used to rapidly compare whole genomes [13]. MUMmer computes alignment scores between sequence pairs with an algorithm primarily based on suffix-tree construction rather than dynamic programming [39]. MUM stands for Maximal Unique Matches which form the anchor points for sequence comparisons, hence the name MUMmer. We generate alignment scores for the *InfA4* (HA), *InfA6* (NA), DENV, and HIV sample datasets using MUMmer, and compute their correlation with ClustalW alignment scores. These are then compared to the highest correlation achieved by the distance methods **D2**, **D2PA**, **KMER**, and **KMOD** (Figure 7.1). For the datasets showing higher standard deviation in alignment scores; *InfA4* (HA), *InfA6* (NA), and HIV, MUMmer correlation is the lowest of all methods while the **KMOD** method achieves the highest correlation. For the DENV dataset, all methods except **KMER** achieve near perfect correlation with ClustalW.

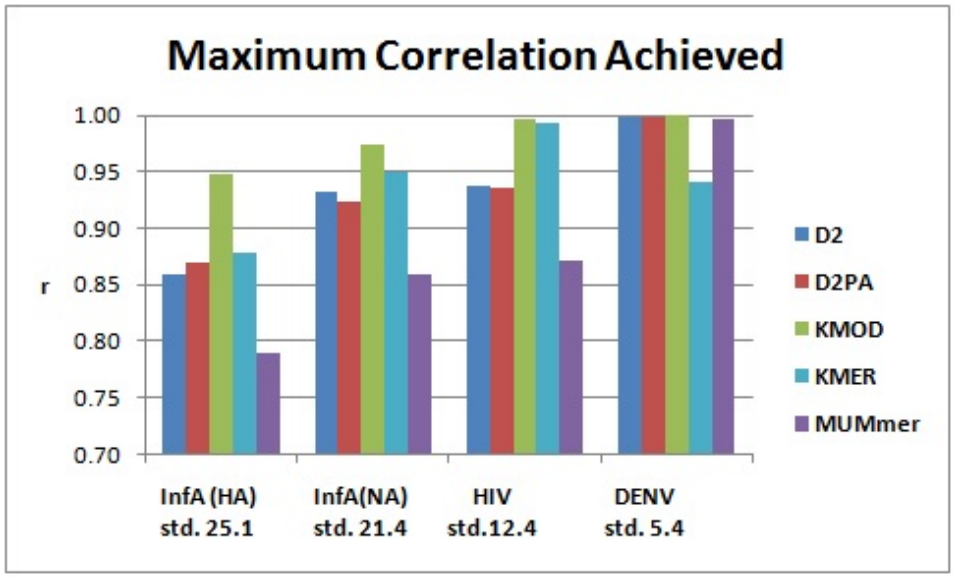

Figure 7.1: Highest correlation achieved with various methods

7.9 Run times

The computational speed of the **KMOD** comparative method is compared with two rapid string comparison programs commonly used in bioinformatics: BLAST and MUMmer. The algorithmic time complexity for comparing two sequences using BLAST is $O(n * m)$ and with MUMmer is $O(n + m)$ [12] where n and m are sequence lengths. The linear-time description of MUMmer is based on the assumption that several large stretches of aligned regions exist between the two sequences being compared [15].

ClustalW is also included in these comparisons. ClustalW alignments involve more than pairwise string comparisons as they compute both multiple sequence alignments and all pairwise alignments on a given set of sequences. We include run times as a general reference. It is noteworthy that ClustalW used 17.5 days to process a dataset of 4,000 short viral sequences, and was unable to complete the processing of a set of five bacterial sequences of approximately 1.6 million base pairs each after two months.

To compute actual run times, a varied number of randomly selected, *InfA1* sequences are used as a basis for comparison. Real run times are calculated on Dual Core Opterons running at 2.6GHz with 8GB of RAM. Times are reported in Table 7.2. The **KMOD** algorithm implemented with the k-deep prefix tree shows the fastest run times for all sequence sets with the largest dataset of 4,000 sequences requiring under five hours. A BLAST comparison on the same dataset requires almost five days. The MUMmer application aborts using larger datasets of 1,000 and 4,000 sequences.

Table 7.2: Run times for all pairwise comparisons using **KMOD**, BLAST and MUMmer. Except for ClustalW, each method label includes its algorithmic complexity.

N	**KMOD** $O(n+m)$	BLAST $O(n*m)$	MUMmer $O(n+m)$	ClustalW
10	1 sec	4 sec	4 sec	16 sec
100	5 sec	7 min	56 sec	17 min
1000	12 min	10.6 hrs	(program abort)	NA
4000	4.9 hrs	4.9 days	(program abort)	17.5 days

7.10 Positional Dependence Metric

This section describes an additional novel k-mer based algorithm that we developed, but did not include in the accuracy testing procedure. It does not outperform any other method included above, but we feel it is worth mentioning here.

In the positional dependence metric, we consider differences in the genomic positions of each k-mer being examined. The position of each k-mer (w) found in a sequence is defined by its nucleotide starting position. Each position is then divided by the total (length of the sequence - k) to give a relative position (r_w). As we compare sequences (S_1,S_2), the relative position of each k-mer in S_1 is compared to the relative position of each matching k-mer in S_2. The distance contributed by that k-mer is the minimum distance found between all its relative positions in S_1 and S_2. The formula is given as:

$$d_{1,2}(w) = \begin{cases} 1 & \text{if } w \in S_1 \text{ and } \notin S_2 \\ min_{1,2} \left| r_w(S_1) - r_w(S_2) \right| & \forall w \in S_1 \cap S_2 \end{cases} \tag{7.9}$$

then $d_{1,2}$ for word length k is:

$$d_{1,2}^k = \frac{\sum_{w_k} d_{1,2}(w)}{|S_1| - k + 1} \tag{7.10}$$

This formula is not symmetric in that in most cases, $d_{1,2} \neq d_{2,1}$. In our application, we forced symmetry by setting each matrix element (i, j) to the minimum of $d_{i,j}$ and $d_{j,i}$.

7.11 Data description

7.11.1 *Influenza A virus*

The *InfA* dataset was obtained from the Influenza Virus Resource [2], and contains 4,228 worldwide, whole viral genomes of multiple subtypes, from several host types, collected between the years 1999-2009. Tables 7.3-7.6 list the number of sequences from each country of origin, subtype, host type, and collection year.

The *InfA* genome exists in eight discontiguous segments denoted by the genes that are encoded for on each of them (Figure 7.2). We divide this *InfA* dataset into eight subsets: *InfA1* (PB2), *InfA2* (PB1), *InfA3* (PA), *InfA4* (HA), *InfA5* (NP), *InfA6* (NA), *InfA7* (M1/M2), *InfA8* (NS1/NS2); distances and accuracy are measured independently for each segment.

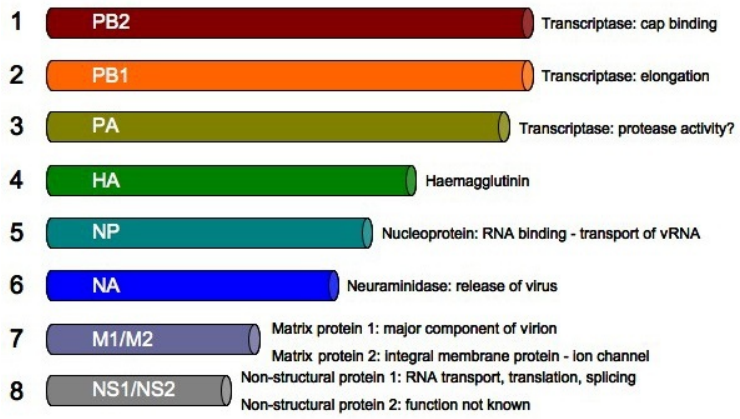

Figure 7.2: The influenza genome is composed of eight discontinuous segments, from [63]

7.11.2 Dengue Virus

The DENV dataset is available from the Virus Variation website [69]. This dataset contains 2,194 whole-genome sequences. All Dengue viral serotypes (DEN-1, DEN-2, DEN-3, DEN-4) are included in this dataset, with distributions shown in Table 7.8. Collection sites are distributed worldwide (Table 7.7) and collection years range from 1944-2009 (Table 7.9). All viral samples stem from human hosts.

7.11.3 HIV

The HIV dataset is housed at the HIV Sequence Compendium [41]. We consider the entire set of 2,377 HIV-1 whole-genome sequences included in the database. Sequences are distributed globally with collection years spanning 1982-2009. See Tables 7.10-7.12 . All viral samples stem from human hosts.

Table 7.3: *InfA* countries of origin

Country	#seqs.	Country	#seqs.	Country	#seqs.
USA	2177	Dominican Republic	12	ElSalvador	2
New Zealand	545	Mongolia	11	United Arab Emirates	1
China	225	Nigeria	10	Guatemala	1
Australia	215	United Kingdom	9	Ukraine	1
Canada	131	Kuwait	9	Croatia	1
Thailand	75	Singapore	9	Kazakhstan	1
Italy	74	Egypt	8	Iran	1
VietNam	74	Spain	7	Niger	1
HongKong	72	CotedIvoire	6	Zambia	1
Indonesia	69	Sudan	6	Slovenia	1
Taiwan	67	France	6	Lebanon	1
Russia	66	Sweden	5	Qatar	1
Nicaragua	63	Afghanistan	5	Czech Republic	1
Japan	61	India	4	Bolivia	1
Mexico	42	Panama	3	Luxembourg	1
Israel	37	Denmark	3	Norway	1
Netherlands	29	Malaysia	3		
Pakistan	22	SaudiArabia	3		
Laos	16	Hungary	3		
South Korea	13	New Caledonia	2		
Germany	12	Colombia	2		

Table 7.4: *InfA* subtypes

Subtype	#seqs.	Subtype	#seqs.	Subtype	#seqs.
H3N2	1428	H6N5	7	H12N4	1
H1N1	1377	H6N8	7	H5N5	1
H5N1	470	H11N2	5	H3N9	1
H7N2	185	H3N5	5	H1N6	1
H9N2	118	H5N7	5	H5N8	1
H3N8	81	H3N1	4	H2N4	1
H7N3	59	H3N3	4	H6N6	1
H1N2	59	H7N7	4	H9N1	1
H6N1	56	H2N2	4	H11N6	1
H4N6	50	H2N1	3	H11N3	1
H5N2	39	H13N9	3	H4N2	1
H7N1	37	H11N8	3	H11N1	1
H10N7	33	H2N9	2	H10N4	1
H6N2	29	H1N3	2	H4N3	1
H2N3	27	Mixed	2	H7N6	1
H3N6	21	H10N8	2	H7N8	1
H11N9	20	H7N9	2	H5N4	1
H4N8	16	H16N3	2	H3N7	1
H5N3	14	H4N7	2	H2N7	1
H12N5	10	H4N5	2		
H8N4	8	H10N3	2		

Table 7.5: *InfA* host types

Host Type	#seqs.
Human	2752
Avian	1194
Swine	128
Environment	114
Unknown	16
Equine	13
Tiger	2
Racoondog	2
Cat	2
Pika	2
Civet	1
Stonemarten	1
Mink	1

Table 7.6: *InfA* collection years

Year	#seqs.
2009	755
2007	656
2005	598
2004	366
2003	323
2000	291
2006	279
2002	273
2008	248
2001	243
1999	196

Table 7.7: DENV countries of collection

Country	#seqs.	Country	#seqs.
Vietnam	944	Nauru	1
USA	293	PuertoRico	1
Nicaragua	221	Honduras	1
Venezuela	218	BurkinaFaso	1
Cambodia	174	India	1
Mexico	83	Jamaica	1
Brazil	79	Dominican Republic	1
Thailand	79	Saint Kitts and Nevis	1
Colombia	56	Belize	1
SriLanka	12	Papua New Guinea	1
None	4	Cook Islands	1
VirginIslands,UnitedStates	3	Samoa	1
Papua New Guinea	3	Mozambique	1
Philippines	2	Ecuador	1
SaintLucia	2	Peru	1
Trinidad and Tobago	2	Anguilla	1
Virgin Islands,British	1	Guyana	1
French Polynesia	1		

Table 7.8: DENV subtypes

Subtypes	#seqs.
DENV-1	1072
DENV-2	643
DENV-3	410
DENV-4	69

Table 7.9: DENV collection years

Year	#seqs.	Year	#seqs.
2007	667	1993	7
2006	378	1991	7
2008	338	2009	6
2001	157	1990	6
2005	154	1986	6
2003	91	1992	5
2004	75	1983	5
1998	60	1988	4
2000	53	1985	3
2002	36	1964	2
1999	29	1944	2
1996	24	1984	1
1995	17	1974	1
1994	17	1973	1
1989	14	1969	1
1997	13	1956	1
1987	13		

Table 7.10: HIV countries of collection

Country	#seqs.	Country	#seqs.	Country	#seqs.	Country	#seqs.
United States	356	Denmark	20	Haiti	5	Mali	2
South Africa	312	Netherlands	18	Israel	5	Paraguay	2
Japan	144	Luxembourg	17	Trinidad and Tobago	5	Romania	2
Cameroon	121	United Kingdom	17	Yemen	5	not defined	2
Thailand	114	Sweden	16	Central African Republic	4	Benin	1
Spain	97	Afghanistan	15	Chad	4	Finland	1
Brazil	95	Senegal	15	Chile	4	Jamaica	1
Kenya	90	Malaysia	13	Dominican Republic	4	Macau	1
Cyprus	75	Ukraine	13	Estonia	4	Malawi	1
Botswana	71	Italy	11	Greece	4	Norway	1
China	65	Russian Federation	11	Hong kong	4	Peru	1
Uganda	63	Uruguay	11	Indonesia	4	Portugal	1
Zambia	50	Belgium	10	Angola, Republic of	3	Somalia	1
Tanzania	48	Nigeria	10	Bolivia	3	Venezuela	1
France	46	Uzbekistan	10	Gabon	3		
Canada	44	Korea,Republic of (South)	9	Germany	3		
Australia	40	Myanmar	9	Niger	3		
Argentina	35	Rwanda	8	Taiwan, Province of China	3		
Congo	34	Coted'ivoire	6	Byelorussianssr	2		
India	34	Kazakhstan	6	Djibouti	2		
Vietnam	33	Saudi Arabia	6	Ethiopia	2		
Ghana	31	Colombia	5	Guinea-bissau	2		
Cuba	20	Georgia	5	Liberia	2		

Table 7.11: HIV subtypes

Subtype	#seqs.	Subtype	#seqs.	Subtype	#seqs.	Subtype	#seqs.
B	702	26_AU	4	AHJU	2	A2CD	1
C	496	28_BF	4	CU	2	A2G	1
01_AE	245	38_BF1	4	DG	2	ACD	1
A1	125	40_BF	4	GKU	2	AD	1
02_AG	69	43_02G	4	K	2	AF2	1
D	60	A1U	4	P	2	AF2G	1
BF1	35	F2	4	213	1	AG	1
01B	32	H	4	225	1	AGU	1
A1D	32	J	4	1819	1	AKU	1
G	31	02D	3	0102A	1	BCF1	1
BF	28	02G	3	01ADF2	1	BCU	1
F1	27	03_AB	3	01AF2U	1	BFG	1
A1C	26	05_DF	3	01C	1	DF1G	1
BC	26	10_CD	3	01DU	1	DO	1
O	25	19_cpx	3	01F2	1	F2KU	1
not defined	23	20_BG	3	02A	1		
42_BF	17	21_A2D	3	02A1U	1		
07_BC	12	26C	3	02AG	1		
35_AD	11	27_cpx	3	02C	1		
BG	11	29_BF	3	02GK	1		
06_cpx	10	31_BC	3	02O	1		
11_cpx	10	34_01B	3	02U	1		
46_BF	10	39_BF	3	06A1	1		
14_BG	9	A1A2D	3	07B	1		
CD	8	A1B	3	17##B	1		
N	8	A1CD	3	26##B	1		
U	8	A1G	3	26CU	1		
12_BF	7	A2	3	27##B	1		
17_BF	7	708	2	29##B	1		
33_01B	7	01BC	2	34##B	1		
02A1	6	02B	2	36##B	1		
08_BC	6	16_A2D	2	44_BF	1		
13_cpx	6	22_01A1	2	72##B	1		
A	6	23_BG	2	74##02D	1		
206	5	32_06A1	2	85##B	1		
01A1	5	36_cpx	2	94##B	1		
09_cpx	5	37_cpx	2	97##B	1		
15_01B	5	47_BF	2	98##B	1		
25_cpx	5	A1A2CD	2	A1CDGKU	1		
45_cpx	5	A1CG	2	A1DHK	1		
209	4	A1DK	2	A1F2	1		
04_cpx	4	A2C	2	A1GHU	1		
18_cpx	4	AC	2	A1GJ	1		
24_BG	4	AGKU	2	A1H	1		

Table 7.12: HIV years of collection

Year	#seqs.	Year	#seqs.
2003	250	1983	32
not defined	223	1985	28
2004	195	1995	28
2002	191	1986	25
2005	190	1994	24
1999	181	1992	23
2000	166	1989	22
2001	161	1991	21
1997	114	2008	19
2006	109	1984	8
1996	108	2009	6
1998	103	1987	5
2007	50	1988	4
1993	49	1982	1
1990	41		

7.12 Results

7.12.1 *InfA*

We observe notable intra-variation in similarity scores in the *InfA* segments. Table 7.13 presents a summary of ClustalW alignment scores per segment. The HA and NA segments show the largest range and standard deviation in alignment scores, 24.2 - 99 and 25.1 and 31 - 99 and 21.4 respectively. This reflects a higher degree of sequence variation than found in other segments. For these segments, the **KMOD** distance method shows the highest degree of correlation with ClustalW full alignment scores over all other methods. Results for the HA segments are shown in Figure 7.6 and Table 7.17. The highest correlation score of 94% is achieved using the **KMOD** method with a word length of 9 or 10. The second highest correlation score of 90% is achieved using the **KMER** method using word lengths of 8 or 9. Results for the NA segment are shown in Figure 7.8 and Table 7.19. For this segment, the **KMER** method also shows the highest correlation of 97% using word lengths $8 - 10$. Again, second highest scoring is the **KMER** method that achieves a correlation of 95% at word lengths $7 - 9$.

In all other segments except for the last (NS1/NS2), **KMER** and **KMOD** both achieve perfect correlation for ranges of word lengths, which begin at 6 or 7. The **D2** and **D2PA** method achieve nearly perfect correlation of 99% beginning at minimum word lengths of $5 - 7$. Graph and tabular results for these segments are shown in Figures 7.3 - 7.5, 7.7, 7.9 and Tables 7.14 - 7.16, 7.18, and 7.20.

In segment NS1/NS2, the **KMOD** method shows the highest correlation of 99% at a minimum word length of 7, while the **KMER** results in a correlation of 98% at a minimum word length of 6. Like the HA and NA segments for which the **KMOD** method also achieves the highest correlation, this segment shows a relatively high range in alignment scores. Table 7.13 shows this segment with the third highest range of scores, following HA with the largest range and NA with the second.

Table 7.13: Alignment score distribution per segment (AVE = average, STD = standard deviation)

segment	MAX	MIN	RANGE	AVE	STD
PB2	99.4	81.6	17.8	86.7	5.4
PB1	99.6	80	19.6	87.0	5.7
PA	99.4	81.2	18.2	87.0	5.3
HA	99	24.2	74.8	52.9	25.1
NP	99.8	80.6	19.2	86.3	5.9
NA	99.4	31	68.4	62.9	21.4
M1/M2	99.8	85.2	14.6	90.4	3.9
NS1/NS2	100	70.2	29.8	83.8	8.3

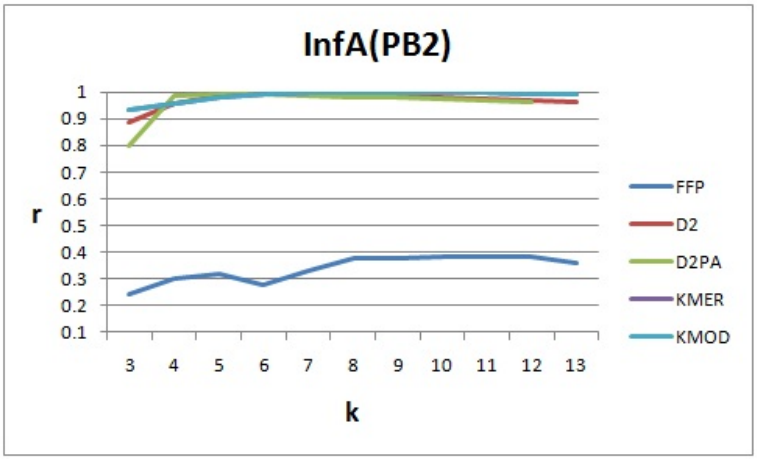

Figure 7.3: Plotted correlation for *InfA1*(PB2)

Table 7.14: Correlations between five distance metrics and ClustalW alignment scores *InfA1*(PB2)

k	FFP	D2	D2PA	KMER	KMOD
3	0.24	0.89	-	0.93	0.93
4	0.30	0.95	0.80	0.96	0.96
5	0.32	0.98	0.98	0.98	0.98
6	0.28	0.99	0.99	0.99	0.99
7	0.33	0.99	0.99	1.00	1.00
8	0.38	0.99	0.99	1.00	1.00
9	0.38	0.98	0.98	1.00	1.00
10	0.38	0.98	0.98	1.00	1.00
11	0.38	0.97	0.97	1.00	1.00
12	0.38	0.97	0.97	0.99	0.99
13	0.36	0.96	0.96	0.99	0.99

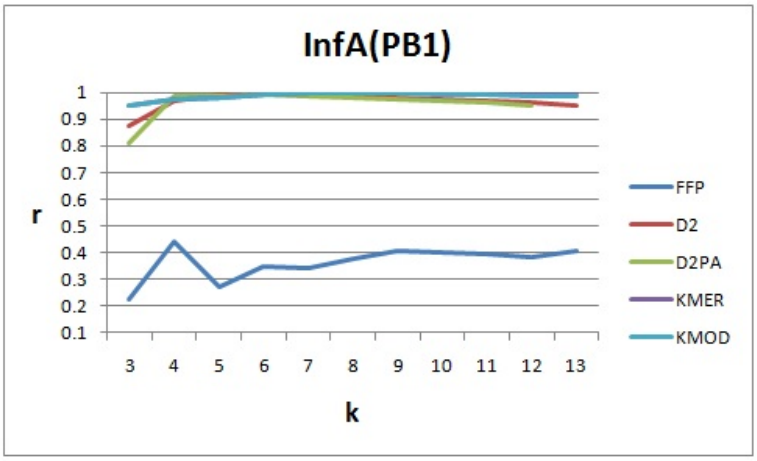

Figure 7.4: Plotted correlation for *InfA2*(PB1)

Table 7.15: Correlations between five distance metrics and ClustalW alignment scores for *InfA2*(PB1)

k	FFP	D2	D2PA	KMER	KMOD
3	0.23	0.87	-	0.95	0.95
4	0.44	0.97	0.81	0.98	0.98
5	0.27	0.99	0.98	0.98	0.98
6	0.35	0.99	0.99	0.99	0.99
7	0.34	0.99	0.99	1.00	1.00
8	0.38	0.98	0.98	1.00	1.00
9	0.41	0.98	0.98	1.00	1.00
10	0.40	0.97	0.97	1.00	0.99
11	0.39	0.97	0.97	1.00	0.99
12	0.38	0.96	0.96	1.00	0.99
13	0.40	0.95	0.95	1.00	0.98

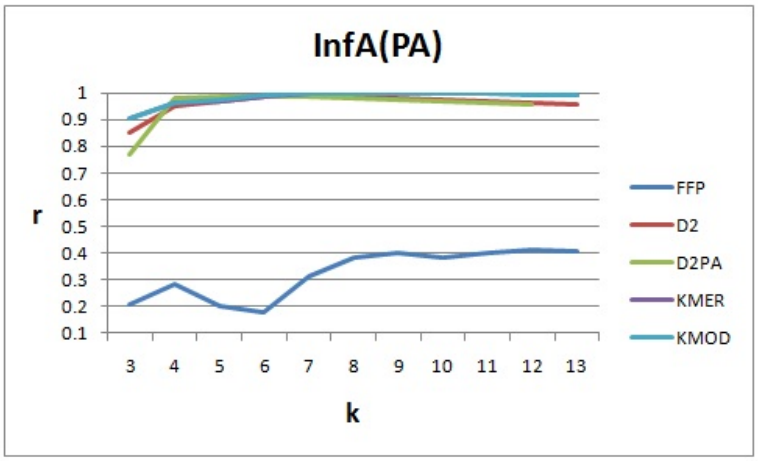

Figure 7.5: Plotted correlation scores for *InfA3*(PA)

Table 7.16: Correlations between five distance metrics and ClustalW alignment scores *InfA3*(PA)

k	FFP	D2	D2PA	KMER	KMOD
3	0.21	0.85	-	0.90	0.90
4	0.28	0.95	0.77	0.96	0.96
5	0.20	0.97	0.98	0.97	0.97
6	0.18	0.98	0.99	0.99	0.99
7	0.31	0.99	0.99	1.00	1.00
8	0.38	0.98	0.98	1.00	1.00
9	0.40	0.98	0.98	1.00	1.00
10	0.38	0.97	0.97	1.00	1.00
11	0.40	0.97	0.97	1.00	1.00
12	0.41	0.96	0.96	1.00	0.99
13	0.41	0.96	0.96	0.99	0.99

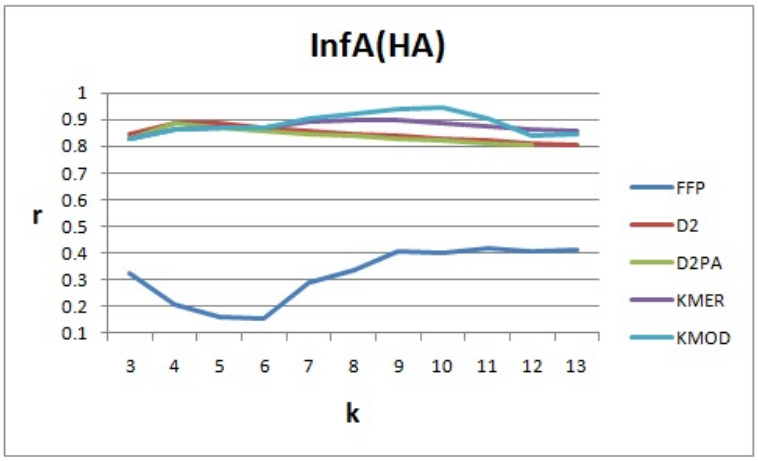

Figure 7.6: Plotted correlation scores for *InfA4* (HA)

Table 7.17: Correlations between five distance metrics and ClustalW alignment scores *InfA4* (HA)

k	FFP	D2	D2PA	KMER	KMOD
3	0.33	0.85	-	0.83	0.83
4	0.21	0.88	0.83	0.86	0.87
5	0.16	0.88	0.89	0.87	0.87
6	0.15	0.87	0.87	0.87	0.87
7	0.29	0.86	0.86	0.89	0.90
8	0.34	0.85	0.85	0.90	0.92
9	0.40	0.84	0.84	0.90	0.94
10	0.40	0.83	0.83	0.89	0.94
11	0.42	0.82	0.82	0.88	0.91
12	0.41	0.81	0.81	0.86	0.84
13	0.41	0.81	0.81	0.86	0.85

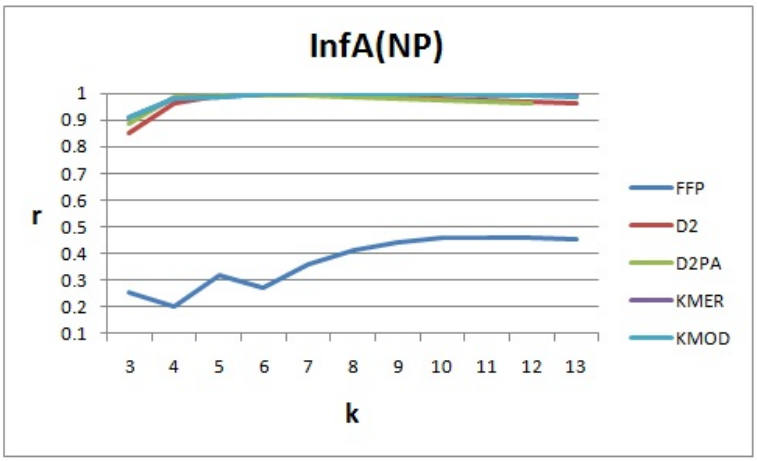

Figure 7.7: Plotted correlation scores for *InfA5*(NP)

Table 7.18: Correlations between five distance metrics and ClustalW alignment scores *InfA5*(NP)

k	FFP	D2	D2PA	KMER	KMOD
3	0.26	0.85	-	0.91	0.91
4	0.20	0.96	0.88	0.98	0.98
5	0.32	0.99	0.99	0.99	0.99
6	0.27	0.99	0.99	1.00	1.00
7	0.36	0.99	0.99	1.00	1.00
8	0.41	0.99	0.99	1.00	1.00
9	0.44	0.99	0.99	1.00	1.00
10	0.46	0.98	0.98	1.00	1.00
11	0.46	0.98	0.98	1.00	0.99
12	0.46	0.97	0.97	1.00	0.99
13	0.46	0.96	0.96	0.99	0.99

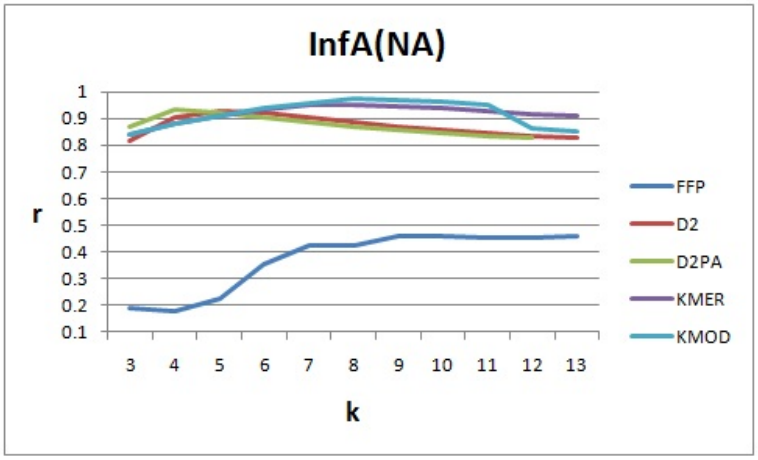

Figure 7.8: Plotted correlation for *InfA6*(NA)

Table 7.19: Correlations between five distance metrics and ClustalW alignment scores *InfA6*(NA)

k	FFP	D2	D2PA	KMER	KMOD
3	0.19	0.82	-	0.84	0.84
4	0.18	0.90	0.87	0.88	0.88
5	0.23	0.93	0.93	0.91	0.91
6	0.35	0.92	0.92	0.93	0.94
7	0.42	0.90	0.90	0.95	0.96
8	0.43	0.89	0.89	0.95	0.97
9	0.46	0.87	0.87	0.95	0.97
10	0.46	0.86	0.86	0.94	0.97
11	0.46	0.85	0.85	0.93	0.95
12	0.46	0.84	0.84	0.92	0.86
13	0.46	0.83	0.83	0.91	0.85

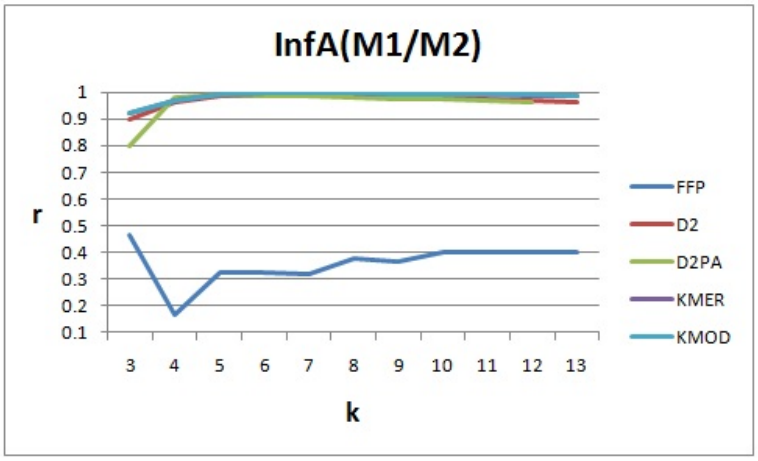

Figure 7.9: Plotted correlation for *InfA7*(M1/M2)

Table 7.20: Correlations between five distance metrics and ClustalW alignment scores *InfA7*(M1/M2)

k	FFP	D2	D2PA	KMER	KMOD
3	0.47	0.90	-	0.92	0.92
4	0.17	0.96	0.80	0.97	0.97
5	0.32	0.98	0.98	0.99	0.99
6	0.32	0.99	0.99	1.00	1.00
7	0.32	0.99	0.99	1.00	1.00
8	0.38	0.98	0.98	1.00	1.00
9	0.37	0.98	0.98	0.99	0.99
10	0.40	0.98	0.98	0.99	0.99
11	0.40	0.97	0.97	0.99	0.99
12	0.40	0.97	0.97	0.99	0.99
13	0.40	0.96	0.96	0.99	0.99

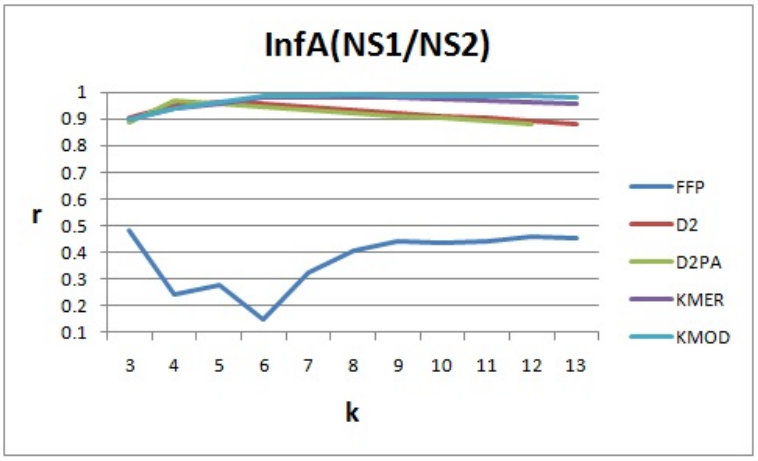

Figure 7.10: Plotted correlation for *InfA8*(NS1/NS2)

Table 7.21: Correlations between five distance metrics and ClustalW alignment scores *InfA8*(NS1/NS2)

k	FFP	D2	D2PA	KMER	KMOD
3	0.48	0.91	-	0.90	0.90
4	0.24	0.95	0.89	0.94	0.94
5	0.28	0.96	0.97	0.96	0.96
6	0.15	0.96	0.96	0.98	0.98
7	0.32	0.94	0.94	0.98	0.99
8	0.41	0.93	0.93	0.98	0.99
9	0.44	0.92	0.92	0.98	0.99
10	0.44	0.91	0.91	0.97	0.99
11	0.44	0.90	0.90	0.97	0.99
12	0.46	0.89	0.89	0.96	0.98
13	0.45	0.88	0.88	0.96	0.98

7.12.2 DENV

ClustalW alignment scores in this dataset show a relatively low range (22.8) and standard deviation (5.4) (Table 7.22). For this dataset, three methods; **D2**, **D2PA**, and **KMOD** all show perfect correlation with alignment scores. The minimum word lengths at which perfect correlation was achieved were 5, 6 and 8, respectively. Figure 7.11 and Table 7.23 show correlation scores achieved by each method.

Table 7.22: DENV ClustalW alignment score distribution

MAX	MIN	RANGE	AVE	STD
97	74.2	22.8	85.2	5.4

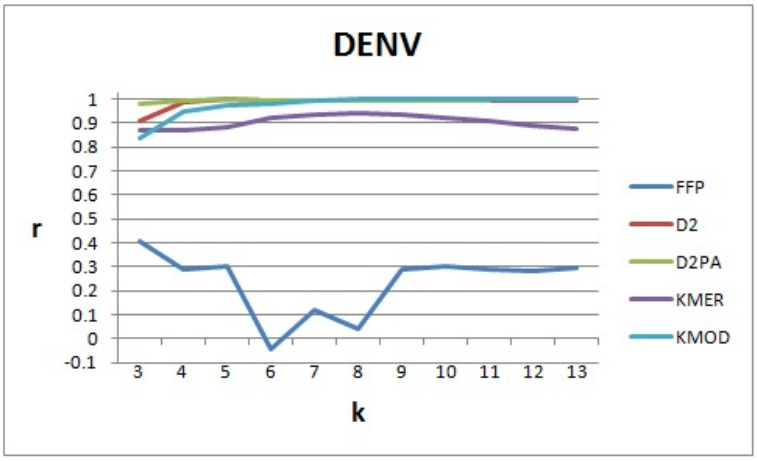

Figure 7.11: Plotted correlation for DENV

Table 7.23: Correlations between five distance metrics and ClustalW alignment scores for DENV

k	FFP	D2	D2PA	KMER	KMOD
3	0.41	0.91	-	0.87	0.84
4	0.29	0.99	-	0.87	0.95
5	0.30	1.00	0.98	0.89	0.97
6	-0.04	1.00	1.00	0.92	0.98
7	0.12	1.00	1.00	0.93	0.99
8	0.04	1.00	1.00	0.94	1.00
9	0.29	1.00	1.00	0.93	1.00
10	0.30	1.00	1.00	0.92	1.00
11	0.29	0.99	0.99	0.91	1.00
12	0.28	0.99	0.99	0.89	1.00
13	0.30	0.99	0.99	0.88	1.00

7.12.3 HIV

Alignment scores in this dataset show a moderately wide range (32.6) and standard deviation (12.4) (Table 7.24). Only one distance method, **KMOD**, achieves perfect correlation with alignment scores. This was found at a minimum word length of 9. The **KMER** method shows the second highest correlation of 99% beginning at a word length of 8. Figure 7.12 and Table 7.25 show correlations of each method.

Table 7.24: HIV ClustalW alignment score distribution

MAX	MIN	RANGE	AVE	STD
99.2	66.6	32.6	79.0	12.4

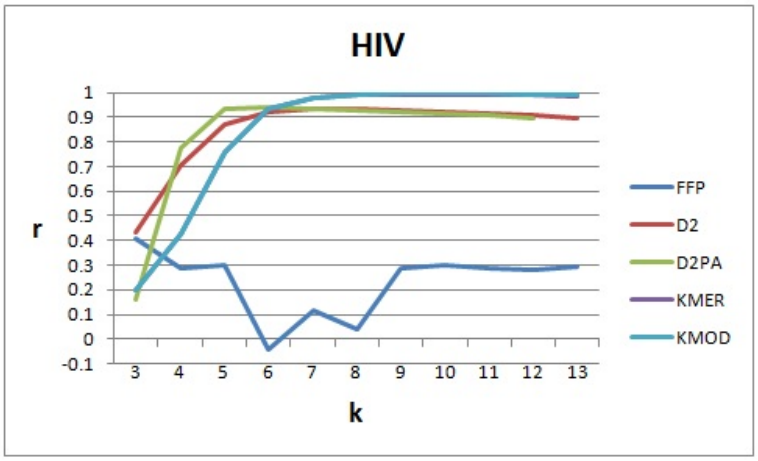

Figure 7.12: Plotted correlation for HIV

Table 7.25: Correlations between five distance metrics and ClustalW alignment scores for HIV

k	FFP	D2	D2PA	KMER	KMOD
3	0.13	0.44	-	0.20	0.20
4	-0.01	0.71	0.16	0.43	0.43
5	0.10	0.87	0.78	0.76	0.76
6	0.02	0.92	0.93	0.93	0.93
7	0.06	0.94	0.94	0.98	0.98
8	0.05	0.94	0.94	0.99	0.99
9	0.05	0.93	0.93	0.99	1.00
10	0.05	0.92	0.92	0.99	1.00
11	0.05	0.91	0.91	0.99	1.00
12	0.11	0.91	0.91	0.99	0.99
13	0.11	0.90	0.90	0.99	0.99

Chapter 8

Graph Theoretic Approaches to Modeling Disease Networks

8.1 Overview

This part of our research focuses on the development of graph models to represent disease transmission using the distance metrics developed and discussed in Chapter 6 and Chapter 7. The basic approach to our method is that each sequence is represented as a graph vertex and vertices are connected by edges only if they represent very similar sequences. We specify the degree of similarity by a distance threshold developed below.

A fast, alignment-free string comparison method is used to compute distance scores for all sequence pairs in the dataset being examined. Using one of these fast methods allows our presented method to accommodate very large subspecies viral data sets. Here we use the **KMOD** metric described in Chapter 7, as it results in the highest correlation with ClustalW pairwise alignment scores for all data sets tested in Chapter 7. This allows that the method presented might accommodate the computation required to incorporate growing subspecies viral data sets.

Using a graph to represent relationships among sequences opens up a number of possible analytical approaches based on existing graph theory and statistical methods. Here, we examine edges across different sequence types by using a concept called 'mixing patterns' [58]. Mixing patterns describe the probability of edges existing between different sequence types in a given network.

Examining patterns of similarity or dissimilarity within and across sequence types, such as place of origin and/or host type, enables us to examine cross-species transmission (i.e. host jumping) and how genotypes are carried over from year to year. The method and analysis presented here try to address some of the current needs for the examination of global influenza circulation [56, 67].

In this chapter, we first describe our general approach, and then apply this method to a complex *Influenza A virus* data set.

8.2 General method to build and examine graphs

8.2.1 Distance Matrix

In the first step, all pairwise distance scores for the entire sequence set are computed using a fast, alignment-free comparison method as presented in Chapter 7. Recall that these methods compute similarity/distance metrics for large datasets with small computational effort. The first step results in an N x N distance matrix computed using the **KMOD** metric (N= number of sequences in a dataset). A small example of this is depicted in Figure 8.1.

8.2.2 Incidence Matrix

From the distance scores computed in the first step, we generate an edge, or incidence matrix. An incidence matrix I is a symmetric NxN matrix, in which a "1" as element I_{ij} represents an edge between vertex^i and vertex^j, and a "0" in I_{ij} denotes no connection between vertex(i) and vertex(j). A graph is defined by its set of vertices and edges representing all connections between any two vertices. Thus, the graph is generated from the incidence matrix.

Distance Thresholds

Distance matrices are converted to incidence matrices by a Boolean conversion. Each distance that is less than or equal to a specified threshold is assigned a value of 1, indicating a similarity of a certain degree. All other distances are converted to 0.

Figure 8.1 illustrates this.

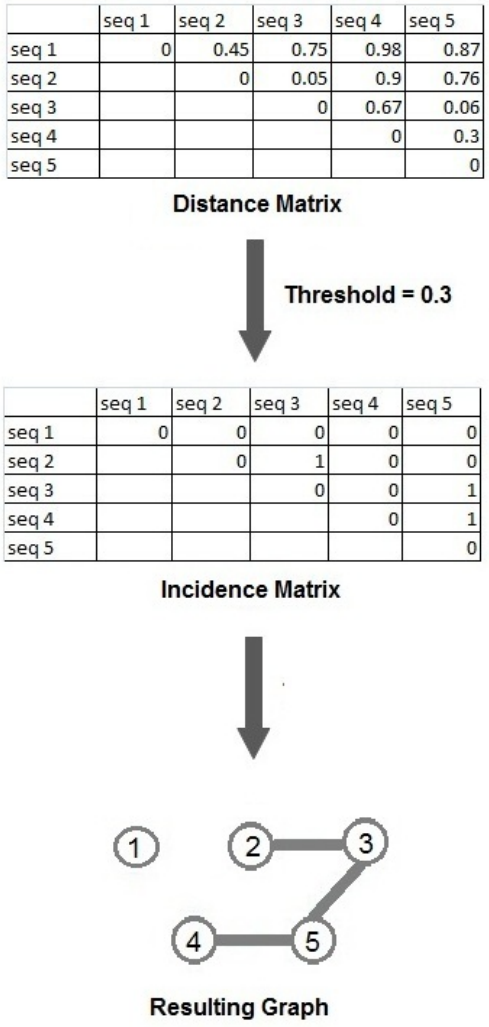

Figure 8.1: Once a distance matrix is generated, it is converted to an edge matrix by instating edges between sequences with distance scores less than or equal to a user-specified threshold. Here, the threshold is 0.30. The resulting graph is shown at the bottom of the image, in which each vertex represents a sequence, and an edge between vertices represents a 'similar' relationship

A distance threshold is a mechanism for labeling two sequences as "similar". If a pairwise distance measure is less than the specified threshold, the two sequences are indicated as being similar to a certain, acceptable degree. Such a threshold can be used for any distance metric (**KMOD** or otherwise) and is commonly used

in applications such as clustering methods to determine when elements are similar enough to be considered elements of the same cluster.

Developing a biologically meaningful distance (or similarity) threshold is non-trivial. We use the ClustalW pairwise similarity scores as a guide to computing meaningful distance thresholds. For a review on ClustalW similarity scores, please see Chapter 7. We first select t_C, a ClustalW pairwise alignment score threshold. We then find distance measures of the **KMOD** distance metric that equate to at least a ClustalW score of t_C.

Due to the computational limitations based upon sequence length and number as described in Chapter 3, it is not feasible to compute the ClustalW scores on all sequence pairs in many datasets of interest. Thus we sample 10% of our datasets to approximate at what values the **KMOD** (or other distance measure) results in at least ClustalW scores of t_C. The set of pairs (**KMOD** distance, ClustalW) is ordered to detect the maximum **KMOD** distance that corresponds to a ClustalW score of t_C. This maximum **KMOD** distance is selected as the distance threshold. Figure 8.2 illustrates this concept for a range of threshold values.

In our analysis, distances are rounded up to the third significant digit to account for floating point errors and minor score discrepancies beyond the third significant digit (this is not illustrated in Figure 8.2).

Distance Threshold Performance

As we are using only a small subset of our data to determine a distance threshold, it is necessary to assess the accuracy of our choice of threshold. We do this by calculating the sensitivity and specificity of each threshold. Sensitivity and specificity are statistical measures used to determine the accuracy of a binary classification, where each sample is assigned to one group or another [1]. Here we determine whether distance scores equal to or less than the specified threshold indicate accurately ClustalW pairwise alignment scores of at least t_C.

Sensitivity gives a measure of the number of alignment scores correctly identified

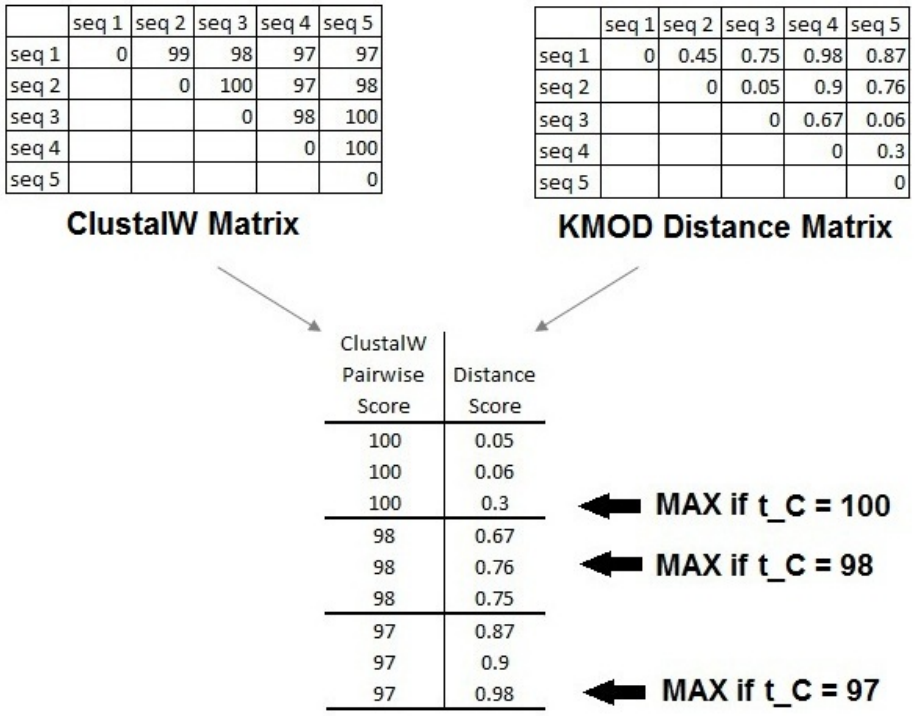

Figure 8.2: ClustalW alignment scores and distance matrices are computed for a set of sequences. The maximum distance value found for a given alignment score of t_C is then selected as a threshold value

by using the specified threshold t_K. A threshold with perfect sensitivity would correctly identify all distance scores corresponding to at least ClustalW alignment scores of t_C. Specificity provides a measure of the exclusivity of the threshold being used. A threshold with perfect selectivity would not mistakenly select any distance scores corresponding with alignment scores of less than t_C.

Sensitivity is calculated here as:

$$\text{sensitivity} = \frac{\text{number of } \textbf{True Positives}}{\text{number of } \textbf{True Positives} + \text{number of } \textbf{False Negatives}} \quad (8.1)$$

where:

True Positives: Sequence pairs with distance scores $\leq t_K$ and ClustalW alignment scores $\geq t_C$

False Negatives: Sequence pairs with distance scores $>t_K$ and ClustalW alignment scores $\geq t_C$

Selectivity is calculated here as:

$$\text{selectivity} = \frac{\text{number of } \textbf{True Negatives}}{\text{number of } \textbf{True Negatives} + \text{number of } \textbf{False Positives}} \quad (8.2)$$

where:

True Negatives: Sequence pairs with distance scores $> t_K$ and ClustalW alignment scores $< t_C$

False Positives: Sequence pairs with distance scores $\leq t_K$ and ClustalW alignment scores $< t_C$

In general, there is a tradeoff in selectivity and sensitivity. Recall that perfect sensitivity does not miss anything, while perfect selectivity does not mistakenly accept anything. A more stringent threshold yields higher selectivity but may yield lower sensitivity. Thresholds may be adjusted by the researcher to suit desired goals. For example, if it is crucial that a test has no false positives, then selectivity would be a more important measure to ascertain accuracy for.

The overall accuracy of a given threshold may be assessed using bounds derived from selectivity and sensitivity scores:

PERFECT if selectivity = 1 and sensitivity = 1
EQUIVALENT TO RANDOM GUESS if sensitivity = (1 - selectivity)
WORSE THAN RANDOM GUESS if sensitivity < (1 - selectivity)
PERFECTLY INCORRECT if sensitivity = 0 and selectivity = 0

A full review of these concepts may be found in [23].

We compute the sensitivity and selectivity measures using a second subset of size 10% of our entire data set, which is mutually exclusive of our first training data subset that was used to determine the k-mer distance metric threshold t_K. We again compute a distance matrix using the **KMOD** method and all pairwise alignment scores using ClustalW. We use these values and Equations 8.1 and 8.2 to assess the performance of our threshold t_K.

8.2.3 Graph connectivity

After determining that the accuracy of the distance threshold is acceptable using the selectivity and sensitivity measures, we use the resulting incidence matrix to represent a graph. Our next step is to examine the graph's connectivity patterns.

Here we examine the occurrence of edges between different vertex types that can indicate transmission across (or within) these types. Types of sequences might include country of origin, continent of origin, organism, type of host, date of collection, etc., or any combination of these. For example, we might be interested in the occurrence of all edges between Human and Avian vertices in North America during the years 1990-1995.

In addition, we examine mixing patterns [58] among vertex types. Mixing patterns describe the probability of edges existing between different types within the given network. Specifically, for each vertex of type A in a network, we compute the

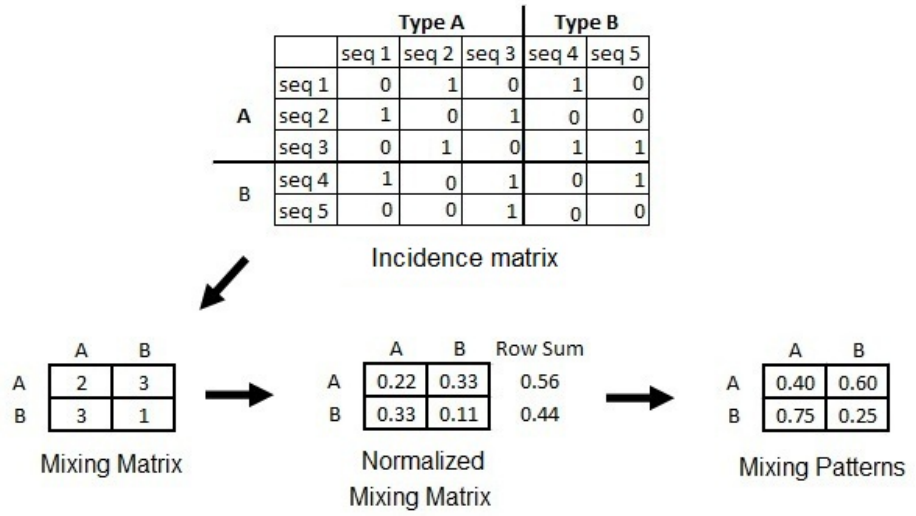

Figure 8.3: Assuming two vertex types exist (A and B), the number of edges between each type is added to generate a mixing matrix. This mixing matrix is then normalized by its total number of edges. To compute mixing patterns, the sum of each row is computed, and each element is divided by its corresponding row sum. Thus, given a vertex of type A, the probability that it is connected to a vertex of type B ($P(B|A)$) is 0.6. By the definition of $P(B|A)$, it is clear that the final mixing pattern matrix is asymmetric.

conditional probability that its neighbor is of type B, i.e., $P(B|A)$.

To examine mixing patterns, a mixing matrix $\mathbf{E}$ is generated in which E_{AB} contains the number of edges connecting vertices of type A to vertices of type B. A normalized mixing matrix $\bar{\mathbf{E}}$ is then derived where:

$$\bar{\mathbf{E}} = \frac{\mathbf{E}}{\|\mathbf{E}\|}$$

$\|\mathbf{E}\|$ represents the sum of all elements in $\mathbf{E}$. $P(B|A)$ for each vertex type A and all neighbor types B can then be computed as $P(B|A) = \bar{E}_{ab} / \sum_b \bar{E}_{ab}$, as described in [58].

Figure 8.3 illustrates an example of mixing pattern calculation.

Examining graph connectivity as described above allows us to approximate transmission among and between different vertex types.

8.3 Methodology applied to *Influenza A Virus*

Here we present our approach applied to a large set of publicly available whole-genome *Influenza A virus* sequences from Influenza Virus Resource [2]. This dataset contains 4,228 worldwide, whole viral genomes of multiple subtypes, from several host types, collected between the years 1999-2009, across 58 countries. Tables in Section 7.11.1 list the number of sequences from each country of origin, subtype, host type, and collection year.

Applying the **KMOD** method on RNA viruses such as *Influenza A virus* is very efficient, as these viruses mutate rapidly [32, 67] and the sequence length is relatively short. Recall that short sequences reduce computational requirements. Furthermore, high mutation rates of RNA viruses create potentially traceable micro-evolutionary pathways through sequence comparisons [42], which we hope to identify.

8.3.1 *InfA* incidence matrices

In Chapter 7, it was determined that distance scores using the **KMOD** comparative method show the best overall correlation with ClustalW pairwise alignment scores for all RNA viral datasets tested. The optimal word length (k value) across all datasets is nine. Recall that the *InfA* genome is composed of eight segments (Fig. 7.2). We begin by computing distance matrices of all pairwise distance scores for each segment in the *InfA* dataset using the **KMOD** distance score with a word length of nine. Note that by definition of **KMOD** these distance matrices are symmetric.

8.3.2 *InfA* distance threshold performance

Because the *InfA* dataset is composed of eight distinct segments, we generate eight individual graphs from eight individual distance thresholds. Each **KMOD** distance threshold t_K is derived from sequence subsets of each segment, yielding eight thresholds $(t_{K1}, .., t_{K8})$.

We estimate **KMOD** distance score values that correspond to ClustalW alignment scores of 97% and 96% for specific segments. We base these choices on the distribution of ClustalW alignment scores for each segment, shown in Figures 8.4 - 8.11. For each segment, a natural break in score distribution occurs at the 96% or 97% score. Thresholds of 97% are applied to segments 1-3, 7, and 8. Thresholds of 96% are applied to segments 4-6.

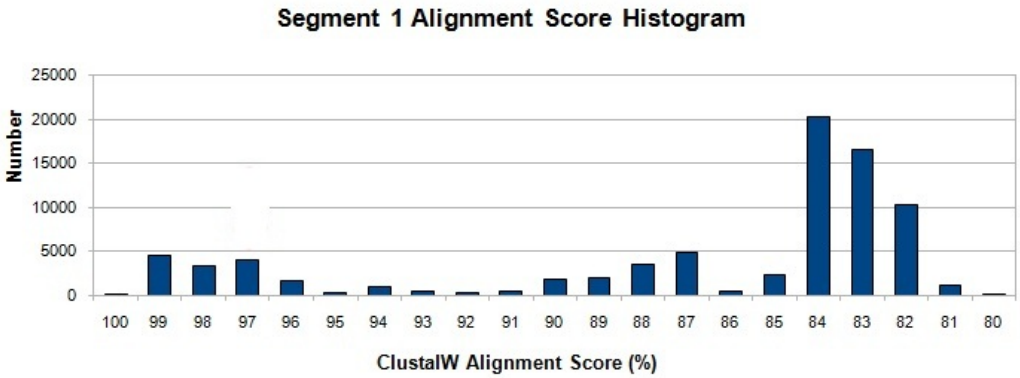

Figure 8.4: Histogram of ClustalW alignment scores for *InfA1*

Using approximately one-tenth of our entire dataset, 400 sequences are selected from each segment specific dataset. For each segment specific subset of size 400, a ClustalW alignment matrix is computed as well as a distance matrix using the **KMOD** distance metric. Thresholds are derived in the manner described in earlier sections.

8.3.3 *InfA* accuracy assessment

Table 8.1 lists sensitivity and selectivity scores per segment. All thresholds show near perfect sensitivity. All thresholds also show near perfect selectivity, with no scores less than 0.98, where 1 is a perfect sensitivity score. Table 8.1 also show the number of True Positives, True Negatives, False Positives and False Negatives counted for each threshold.

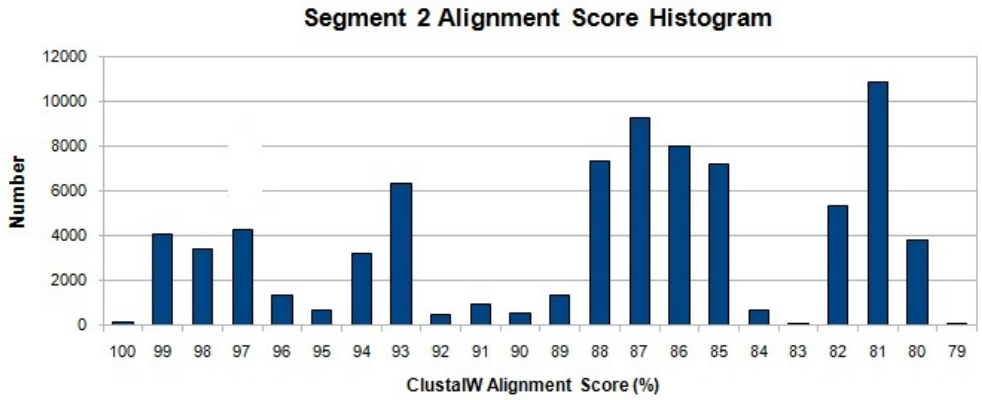

Figure 8.5: Histogram of ClustalW alignment scores for *InfA2*

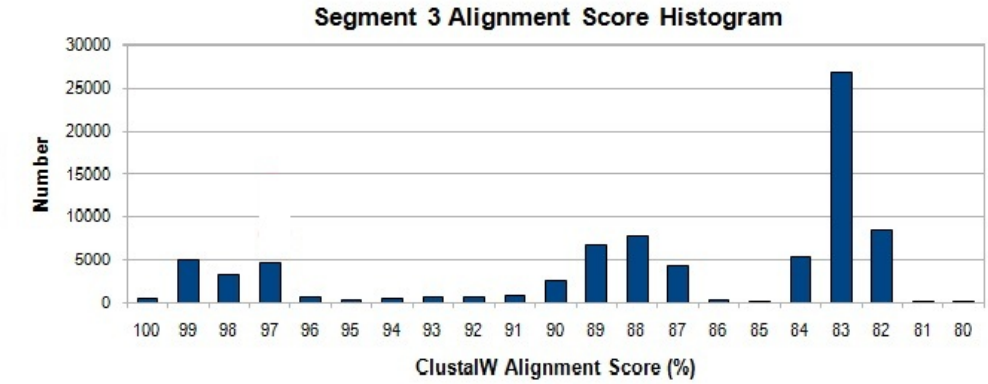

Figure 8.6: Histogram of ClustalW alignment scores for *InfA3*

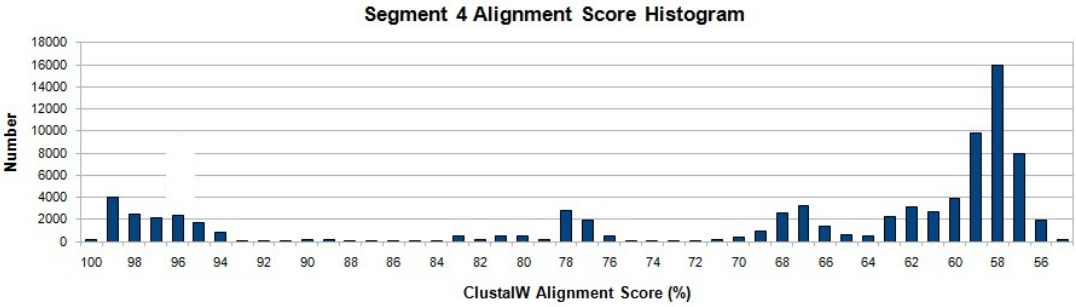

Figure 8.7: Histogram of ClustalW alignment scores for *InfA4*

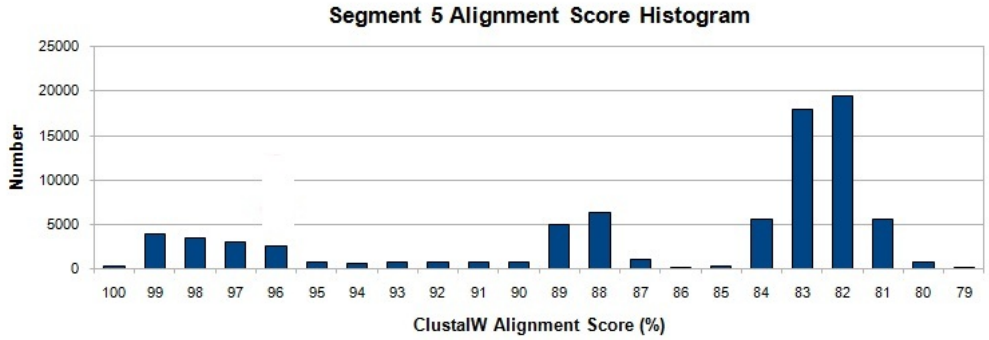

Figure 8.8: Histogram of ClustalW alignment scores for *InfA5*

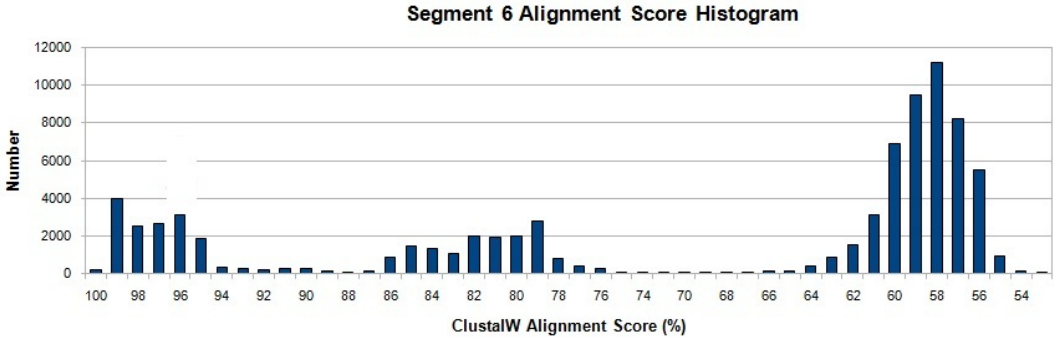

Figure 8.9: Histogram of ClustalW alignment scores for *InfA6*

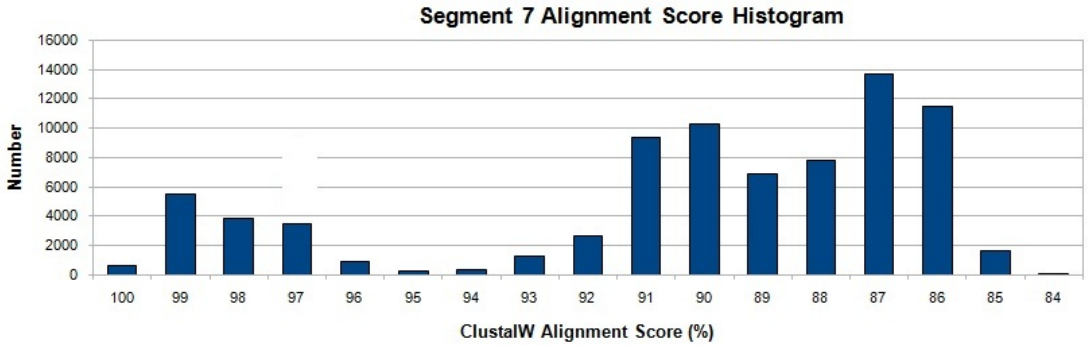

Figure 8.10: Histogram of ClustalW alignment scores for *InfA7*

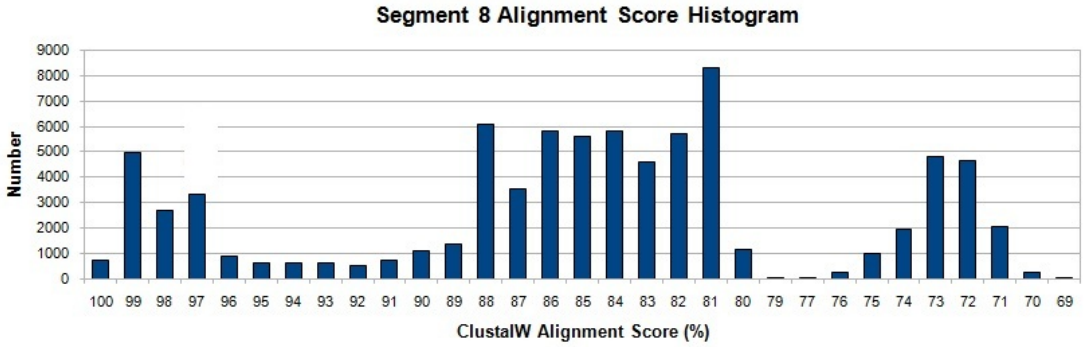

Figure 8.11: Histogram of ClustalW alignment scores for *InfA8*

Table 8.1: Sensitivity and selectivity of threshold scores per segment

Threshold	Sens.	Sel.	True POS	False POS	True NEG	False NEG
t_{K1}	1	0.990	12,221	757	66,822	0
t_{K2}	1	0.995	12,025	352	67,422	1
t_{K3}	1	0.993	13,510	443	65,846	0
t_{K4}	1	0.980	11,246	1,338	67,216	0
t_{K5}	1	0.995	13,372	352	66,076	0
t_{K6}	1	0.980	12,489	1,321	65,990	0
t_{K7}	1	0.993	13,405	453	65,938	4
t_{K8}	1	0.992	11,707	541	67,551	1
AVERAGE	1	0.998				

Table 8.2: Number of edges per segment

Segment	Number of edges	Ave. Degree
1	73,879	9.3
2	37,832	17.9
3	36,472	17.3
4	18,688	8.8
5	31,213	14.8
6	37,536	17.8
7	34,295	16.2
8	73,879	35.0
TOTAL	269,915	

8.3.4 *InfA* resulting graph

The final graph is computed as the sum of all segment specific graphs. Given N whole viral sample sequences composed of disjoint segments 1-8, a summed $N \times N$ graph G is created by summing all pairwise edges across all segments:

$$\forall i, j \in N,\ G_{i,j} = \sum_{seg=1}^{8} g_{seg,i,j} \tag{8.3}$$

The resulting graph constructed from the *InfA* dataset contains 269,915 edges. The number of edges per segment are listed in Table 8.2. The degree(k) of a vertex is the total number edges connecting to it, whereas the average degree ($\overline{k}$) is the average of all vertex degrees in a graph. The average degree for each segment is also reported in Table 8.2. Segments 1 and 4 show the lowest average degree and segment 8 shows the highest average degree.

8.3.5 *InfA* cross-type edges and mixing patterns

Here we examine edges between host types and collection years. cross-type edges might indicate transmission across hosts and the persistence of genotypes across years.

Pinpointing which edges indicate transmission across hosts addresses 'host jumping'. This is a characteristic of *Influenza A virus* which has enabled the emergence of highly pathogenic strains such as the recent H5N1 bird flu and the Spanish flu in

1918 [86]. Examining the persistence of genotypes across years addresses the question of whether new influenza strains emerge and circulate globally each year, or whether they remain in localized pockets and re-emerge periodically [57]. We do not address these questions directly in this research. Instead, we simply illustrate how the described method can be used to address specific questions regarding disease transmission.

We quantify the number of inter-type edges in detail and for certain cross-type edges, we examine the countries where these edges occur. We also examine which *InfA* segments contribute the most to edges across various types. In addition, we examine mixing patterns [58] across host and year types.

Inter-host edges

The host types of our data set include Human, Domestic Avian, Wild Avian, Swine, Mammal, Environment, or Unknown. The Domestic Avian class includes all viral samples from hosts labeled chicken, turkey, duck, or goose. Samples from hosts labeled by wild bird species, such as "mallard" or "egret" are considered Wild Avian. The Mammal class is broad and includes all non-human and non-swine mammals, including species such as horse, civet and tiger. As described in Section 7.11.1, sequences were obtained during the years 1999-2009 and were collected in 58 different countries.

Table 8.3 displays the number of edges found between different host types in the summed graph G, and the contributions from each segment specific graph ($g_1 - g_8$). The largest number of inter-host edges ($1,094$) are found between the Wild Avian and Domestic Avian groups. The non-human host type showing the highest number of edges with Human is Swine. Links are also found between Human and Domestic and Wild Avian types, however, while these edges only sum to 27, the number of edges between Human and Swine samples is 197.

Table 8.3: Number of edges between host types. All host types are followed by parenthesis containing the number of vertices in the graph of that type

Host Type	Host Type	Edges	g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8
WildAvian (327)	DomAvian (867)	1094	85	100	89	102	151	126	207	234
DomAvian (867)	Environment (115)	645	40	60	52	97	84	97	97	118
Human (2750)	Swine (130)	197	2	16	32	1	46	19	30	51
Avian (867)	Unknown (16)	146	10	18	12	28	25	26	19	8
WildAvian (327)	Environment (115)	98	6	11	4	10	9	11	26	21
Environment (115)	Unknown (16)	85	5	13	1	15	5	25	15	6
WildAvian (327)	Unknown (16)	50	5	9	4	5	5	6	7	9
Human (2750)	DomAvian (867)	21	1	1	1	1	8	1	4	4
DomAvian (867)	Swine (130)	13	3	0	0	1	0	1	5	3
WildAvian (327)	Human (2750)	6	0	3	0	0	0	0	0	3
WildAvian (327)	Swine (130)	5	0	0	0	0	0	0	2	3
WildAvian (327)	Mammal (23)	2	0	0	0	0	1	0	1	0
DomAvian (867)	Mammal (23)	2	0	0	0	0	0	1	1	0
Environment (115)	Mammal (23)	1	0	1	0	0	0	0	0	0
Swine (130)	Mammal (23)	1	0	0	0	0	0	0	0	1
TOTAL		2366	157	232	195	260	334	313	414	461
% per segment			7%	10%	8%	11%	14%	13%	17%	19%

Figure 8.12 is a heatmap of the conditional probabilities that a vertex of type A has an edge with type B in the overall graph G. Table 8.4 lists the numerical values used to generate the heatmap. Diagonal entries on the heatmap reflect the tendency for edges to be found between identical host types. This image displays the connectedness of Domestic and Wild Avian types, also shown by the high number of edges found between these classes in Table 8.3. Unknown type samples show a high probability of forming edges with both Domestic Avian, as well as Environment vertices. Environment types form higher numbers of edges with Unknown and Domestic Avian types. This image also reflects the relatively high numbers of edges between Swine and Human samples.

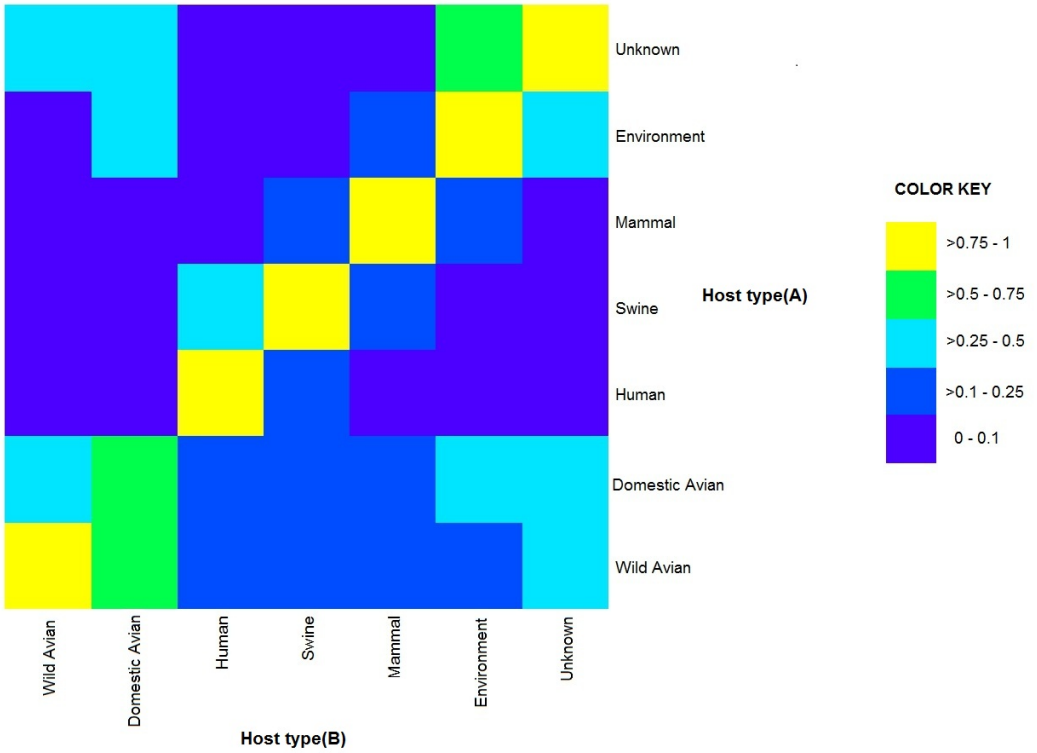

Figure 8.12: Probability that a vertex of host type (A) has an edge with a vertex of host type (B) in the network

Table 8.4: $P(B \mid A)$ for inter-host mixing. Values less than 0.01 are omitted

		B						
		Wild Avian	Domestic Avian	Human	Swine	Mammal	Environment	Unknown
A	Wild Avian	0.35	0.57				0.05	0.03
	Domestic Avian	0.20	0.65				0.12	0.03
	Human			1.00				
	Swine	0.01	0.02	0.27	0.70			
	Mammal	0.03	0.03		0.01	0.92	0.01	
	Environment	0.06	0.37				0.52	0.05
	Unknown	0.15	0.43				0.25	0.18

Edges between host type pairs of particular relevance to host-jumping and global transmission are examined in more detail. These pairs include Human/Swine, Human/Avian (Domestic and Wild), and Domestic Avian/Wild Avian.

Human and Swine samples are connected by 197 edges. The dates and countries of origin of samples forming these links are listed in Table 8.5. In Table 8.5, New Zealand/Human/2000 samples form edges with Canada/Swine/2003 and China/Swine /2003 samples. These edges connecting with New Zealand/Human vertices account for 96% of all Human/Swine edges. Edges are also found between Canada/Human/2005 and Canada/Swine/2005 samples. The contribution of edges from each segment is shown in Table 8.10. While all segments are represented in Human/Swine edges, segments 1 and 4 are the least common whereas segments 5 and 8 are the most common.

Table 8.5: Edges found between Human and Swine samples

Human	Swine	Edges	g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8
NewZealand, Human, 2000 (120)	Canada, Swine,2003 (5)	65	0	0	16	0	23	2	0	24
NewZealand, Human, 2000 (120)	China, Swine,2003 (4)	124	0	15	16	0	23	16	30	24
Canada, Human, 2005 (2)	Canada, Swine,2005 (5)	8	2	1	0	1	0	1	0	3
	TOTAL	197	2	16	32	1	46	19	30	51

Human and Avian (both Wild and Domestic) samples are connected by 27 edges. Most of the edges found are between Human and Domestic Avian. cross-species links all occurred within the same country and within the same year or consecutive years. The countries where host-jumping between Human and Avian species occurred are indicated in this graph and include Canada, the Netherlands, and Thailand. As noted between Human and Swine samples, segments 5 and 8 contribute the most to these cross-species edges.

Table 8.6: Edges found between Human and Avian samples

Human	Avian	Edges	g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8
Canada, Human, 2004 (1)	Canada, D.Avian, 2004 (1)	11	0	0	1	1	3	1	2	3
Canada, Human, 2005 (2)	Canada, D.Avian, 2005 (2)	1	1	0	0	0	0	0	0	0
Netherlands, Human, 2003 (1)	Netherlands, D.Avian, 2003 (1)	2	0	0	0	0	0	0	2	0
Thailand, Human, 2004 (4)	Thailand, D.Avian, 2004 (4)	5	0	0	0	0	5	0	0	0
Thailand, Human, 2006 (1)	Thailand, D.Avian, 2005 (1)	2	0	1	0	0	0	0	0	1
Thailand, Human, 2006 (6)	Thailand, W.Avian, 2005 (6)	6	0	3	0	0	0	0	0	3
	TOTAL	27	1	4	1	1	8	1	4	7

Wild and Domestic Avian samples are connected by 848 edges. These links occur in several countries and during several years. With the exception of two edges between Thailand Avian vertices in 2004 and 2008, all edges are found between vertices of the same year type or consecutive years. Most edges are also across vertices collected in the same country. Tables 8.7- 8.9 display an extended table showing Domestic Avian and Wild Avian vertex types with edges between them. Entries marked with

an * denote vertices from different countries. Inter country edges are found between Canada and the US, China and Russia, Italy and Hungary, Mongolia and Russia, and South Korea and Japan.

Table 8.7: Edges found between Domestic and Wild Avian samples

Avian type	Avian type	Edges	g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8
Australia,D.Avian,2004 (3)	Australia,W.Avian,2004 (2)	9	1	0	2	1	2	1	2	0
Australia,W.Avian,2004 (2)	Australia,D.Avian,2004 (3)	9	1	0	2	1	2	1	2	0
Canada,D.Avian,2000 (2)	Canada,W.Avian,2000 (1)	1	0	0	0	0	0	1	0	0
Canada,D.Avian,2005 (9)	Canada,W.Avian,2005 (6)	1	0	0	0	0	0	0	1	0
Canada,D.Avian,2006 (9)	Canada,W.Avian,2006 (10)	18	1	1	1	2	1	3	4	5
Canada,W.Avian,2001 (3)	Canada,D.Avian,2001 (3)	5	0	1	0	0	1	0	2	1
Canada,W.Avian,2006 (10)	Canada,D.Avian,2006 (9)	18	1	1	1	2	1	3	4	5
* Canada,W.Avian,2006 (10)	USA,D.Avian,2006 (26)	1	0	0	0	0	0	0	1	0
China,D.Avian,2000 (17)	China,W.Avian,2000 (2)	8	0	2	0	0	1	2	1	2
China,D.Avian,2005 (14)	China,W.Avian,2005 (6)	3	0	0	0	0	0	1	2	0
China,D.Avian,2006 (14)	China,W.Avian,2006 (3)	2	0	0	0	0	0	0	0	2
* China,D.Avian,2005 (14)	Russia,W.Avian,2005 (5)	1	0	0	0	0	0	1	0	0
China,W.Avian,2004 (1)	China,D.Avian,2004 (18)	6	1	1	1	1	0	0	1	1
China,W.Avian,2005 (6)	China,D.Avian,2006 (14)	1	0	0	0	0	0	0	0	1
China,W.Avian,2005 (6)	China,D.Avian,2005 (14)	3	0	0	0	0	0	1	2	0
China,W.Avian,2006 (3)	China,D.Avian,2006 (14)	2	0	0	0	0	0	0	0	2
Germany,W.Avian,2006 (1)	Germany,D.Avian,2006 (1)	2	0	0	0	0	1	0	1	0
HongKong,D.Avian,2008 (4)	HongKong,W.Avian,2008 (2)	1	0	0	1	0	0	0	0	0
HongKong,W.Avian,1999 (1)	HongKong,D.Avian,1999 (2)	1	0	0	0	0	1	0	0	0
HongKong,W.Avian,2007 (1)	HongKong,D.Avian,2008 (4)	1	0	0	0	0	0	0	0	1
Indonesia,D.Avian,2004 (4)	Indonesia,W.Avian,2004 (1)	8	1	1	1	1	1	1	1	1
Italy,D.Avian,1999 (21)	Italy,W.Avian,2000 (2)	3	0	0	0	0	2	0	1	0
Italy,D.Avian,1999 (21)	Italy,W.Avian,1999 (1)	9	0	1	1	0	2	0	1	4
Italy,D.Avian,2000 (7)	Italy,W.Avian,2000 (2)	2	0	0	0	0	1	0	1	0
Italy,D.Avian,2006 (4)	Italy,W.Avian,2006 (2)	1	0	0	0	0	0	0	0	1

Table 8.8: Edges found between Domestic and Wild Avian samples (continued)

Avian type	Avian type	Edges	g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8
* Italy,W.Avian,2006 (2)	Hungary,D.Avian ,2006 (2)	1	0	0	0	0	0	0	0	1
Italy,W.Avian,1999 (1)	Italy,D.Avian,1999 (21)	9	0	1	1	0	2	0	1	4
Italy,W.Avian,2004 (1)	Italy,D.Avian,2004 (7)	10	1	1	1	0	3	2	1	1
Japan,D.Avian,2004 (6)	Japan,W.Avian,2004 (1)	5	0	2	1	0	2	0	0	0
Japan,D.Avian,2008 (2)	Japan,W.Avian,2008 (2)	13	2	2	0	2	4	0	2	1
Japan,W.Avian,2004 (1)	Japan,D.Avian,2004 (6)	5	0	2	1	0	2	0	0	0
Japan,W.Avian,2008 (2)	Japan,D.Avian,2008 (2)	13	2	2	0	2	4	0	2	1
Mongolia,D.Avian,2006 (1)	Mongolia,W.Avian,2006 (1)	2	0	0	0	1	0	0	1	0
* Mongolia,D.Avian,2006 (1)	Russia,W.Avian,2006 (4)	1	0	0	0	0	0	0	1	0
* Mongolia,W.Avian,2005 (2)	Russia,D.Avian,2005 (14)	1	0	0	0	0	0	0	1	0
Russia,D.Avian,2005 (14)	Russia,W.Avian,2005 (5)	53	5	5	3	10	12	2	4	12
Russia,D.Avian,2006 (4)	Russia,W.Avian,2006 (4)	8	1	1	1	1	1	1	1	1
Russia,D.Avian,2007 (10)	Russia,W.Avian,2007 (8)	29	2	4	1	5	7	5	2	3
* Russia,W.Avian,2009 (1)	Mongolia,D.Avian,2009 (1)	2	0	0	1	0	0	0	1	0
Russia,W.Avian,2005 (5)	Russia,D.Avian,2005 (14)	53	5	5	3	10	12	2	4	12
Russia,W.Avian,2007 (8)	Russia,D.Avian,2007 (10)	29	2	4	1	5	7	5	2	3
* SouthKorea,D.Avian,2006 (2)	Japan,W.Avian,2007 (1)	1	0	0	0	0	0	1	0	0
Thailand,D.Avian,2004 (11)	Thailand,W.Avian,2004 (3)	2	0	0	2	0	0	0	0	0
Thailand,D.Avian,2005 (14)	Thailand,W.Avian,2005 (14)	166	6	12	15	9	18	19	50	37
Thailand,D.Avian,2007 (1)	Thailand,W.Avian,2008 (1)	1	0	0	0	1	0	0	0	0
Thailand,W.Avian,2004 (3)	Thailand,D.Avian,2008 (4)	2	0	0	2	0	0	0	0	0
Thailand,W.Avian,2005 (14)	Thailand,D.Avian,2005 (14)	166	6	12	15	9	18	19	50	37
USA,D.Avian,1999 (14)	USA , W.Avian,1999 (12)	27	3	4	0	4	4	5	3	4
USA,D.Avian,2000 (12)	USA , W.Avian,2000 (9)	5	0	0	0	0	0	0	4	1
USA,D.Avian,2001 (22)	USA , W.Avian,2001 (14)	119	6	8	7	11	12	11	31	33

Table 8.9: Edges found between Domestic and Wild Avian samples (continued)

Avian type	Avian type	Edges	g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8
USA,D.Avian,2002 (29)	USA,W.Avian,2002 (23)	45	7	1	2	1	4	7	9	14
USA,D.Avian,2003 (10)	USA,W.Avian,2003 (11)	40	5	2	3	2	2	9	4	13
USA,D.Avian,2004 (13)	USA,W.Avian,2004 (10)	26	1	1	2	5	5	2	5	5
USA,D.Avian,2005 (95)	USA,W.Avian,2006 (16)	2	0	0	0	0	2	0	0	0
USA,D.Avian,2005 (95)	USA,W.Avian,2005 (42)	180	14	25	13	20	21	19	36	32
USA,D.Avian,2006 (26)	USA,W.Avian,2006 (16)	43	2	2	2	2	6	4	5	20
USA,D.Avian,2007 (55)	USA,W.Avian,2007 (51)	154	19	18	14	17	24	24	15	23
USA,W.Avian,1999 (12)	USA,D.Avian,1999 (14)	27	3	4	0	4	4	5	3	4
USA,W.Avian,2000 (9)	USA,D.Avian,2000 (12)	5	0	0	0	0	0	0	4	1
USA,W.Avian,2001 (14)	USA,D.Avian,2001 (22)	119	6	8	7	11	12	11	31	33
USA,W.Avian,2002 (23)	USA,D.Avian,2002 (29)	45	7	1	2	1	4	7	9	14
USA,W.Avian,2003 (11)	USA,D.Avian,2003 (10)	40	5	2	3	2	2	9	4	13
USA,W.Avian,2004 (10)	USA,D.Avian,2004 (13)	26	1	1	2	5	5	2	5	5
USA,W.Avian,2005 (42)	USA,D.Avian,2006 (26)	4	0	0	0	0	4	0	0	0
USA,W.Avian,2005 (42)	USA,D.Avian,2005 (95)	180	14	25	13	20	21	19	36	32
USA,W.Avian,2006 (16)	USA,D.Avian,2006 (26)	43	2	2	2	2	6	4	5	20
USA,W.Avian,2007 (51)	USA,D.Avian,2007 (55)	154	19	18	14	17	24	24	15	23
VietNam,D.Avian,2004 (17)	VietNam,W.Avian,2004 (1)	11	0	1	6	1	1	1	0	1
VietNam,D.Avian,2005 (46)	VietNam,W.Avian,2005 (3)	53	7	4	6	5	5	4	11	11
VietNam,W.Avian,2005 (3)	VietNam,D.Avian,2005 (46)	53	7	4	6	5	5	4	11	11
	TOTAL	2089	166	192	162	198	281	242	396	452

Table 8.10 shows the percentage of segments forming links in the three host pairs examined. Segments 5 and 8 contribute the most to edge formation between Human and non-human hosts. Segments 7 and 8 contribute the most to edges between Avian types.

Table 8.10: Percentage of each segment forming inter-host type edges

	g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8
Human/Swine	1%	8%	16%	1%	23%	10%	15%	26%
Human/Avian (Wild,Dom)	4%	15%	4%	4%	30%	4%	15%	26%
Wild Avian/Domestic Avian	8%	9%	8%	9%	13%	12%	19%	22%

Inter-year edges

Table 8.11 shows the number of edges found between samples collected in different years. The majority of edges (95%) are found between pairs of consecutive years. Edge statistics between consecutive years are denoted with a * in Table 8.11. Notably, segment 8 contributes to 70% of all edges.

Table 8.11: Number of edges between years

Year	Year	Edges	g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8
*2007(657)	2008(248)	3034	13	3	5	3	6	1	21	2982
*2005(600)	2004(366)	2792	40	14	20	44	110	202	355	2007
*1999(194)	2000(289)	1132	19	11	10	6	129	114	631	212
*2003(325)	2004(366)	1003	2	7	8	10	22	15	167	772
*2001(242)	2002(273)	444	4	21	4	7	22	17	200	169
*2000(289)	2001(242)	375	0	1	1	1	1	1	10	360
*2007(657)	2006(279)	335	7	18	20	27	60	5	81	117
2000(289)	2003(325)	189	0	15	32	0	46	18	30	48
*2005(600)	2006(279)	175	4	8	7	2	25	8	98	23
2005(600)	2003(325)	122	0	0	0	0	0	0	0	122
2005(600)	2007(657)	84	0	0	0	0	0	0	83	1
*2003(325)	2002(273)	46	11	1	5	3	4	1	13	8
*2008(248)	2009(755)	37	0	0	0	0	0	0	26	11
1999(194)	2003(325)	30	0	0	0	0	0	0	0	30
2002(273)	2004(366)	10	1	1	1	1	1	1	2	2
2003(325)	2001(242)	7	0	1	0	0	0	0	6	0
2004(366)	2006(279)	7	1	0	0	0	0	0	6	0
2004(366)	2008(248)	7	1	0	3	0	0	0	2	1
1999(194)	2005(600)	3	0	0	0	0	3	0	0	0
2007(657)	2004(366)	3	0	0	0	0	0	0	3	0
2003(325)	2007(657)	1	0	0	0	0	0	0	0	1
	TOTAL	9836	103	101	116	104	429	383	1734	6866
		% per segment	1%	1%	1%	1%	4%	4%	18%	70%

Figure 8.13 is a heatmap generated from the conditional probabilities computed for inter-year mixing. This image displays the tendency for edges to be primarily found between samples collected during the same year or those collected in consecutive years. Table 8.12 displays the values used to generate the image.

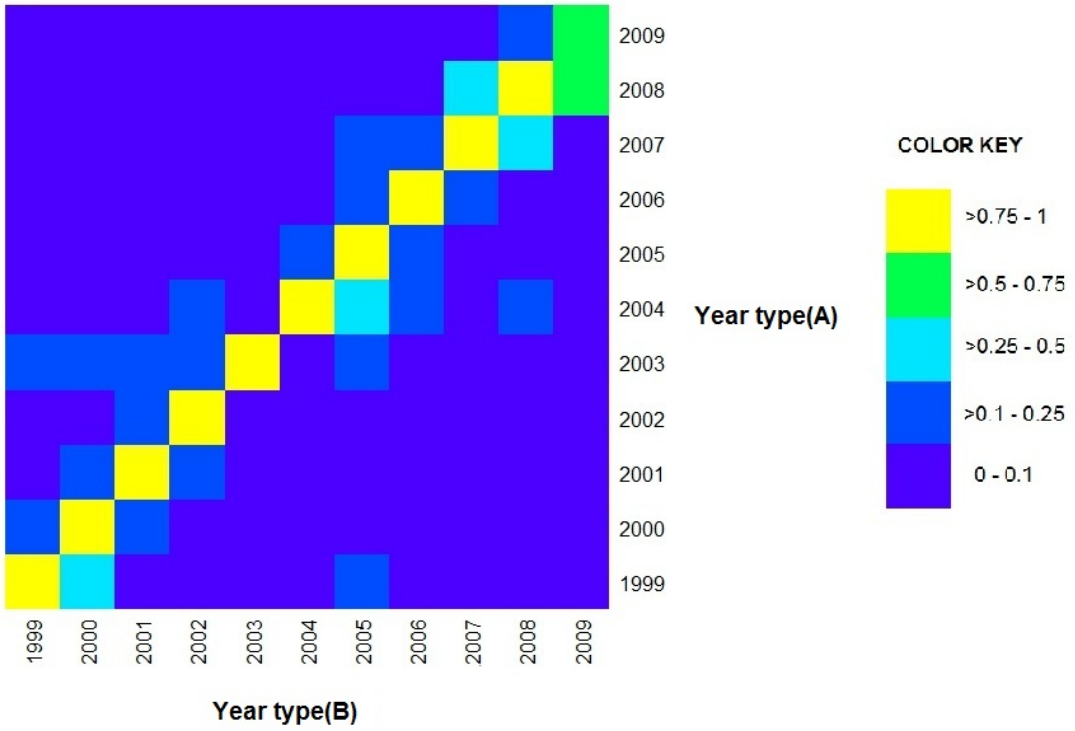

Figure 8.13: Probability that a sample collected in year (A) has an edge with a sample collected in year (B) in the network

Table 8.12: $P(B \mid A)$ for inter-year mixing. Values less than 0.001 are not shown

	B										
	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009
	1999	0.801	0.194			0.005		0.001			
	2000	0.076	0.885	0.025		0.013					
	2001		0.047	0.896	0.056	0.001					
	2002			0.047	0.947	0.005	0.001				
	2003	0.001	0.008		0.002	0.941	0.043	0.005			
A	2004				0.001	0.051	0.804	0.143			
	2005					0.009	0.215	0.755	0.014	0.006	
	2006						0.003	0.077	0.772	0.148	
	2007							0.005	0.018	0.813	0.164
	2008						0.001			0.264	0.732
	2009										1.00

8.4 Estimating number of mutations

In this section, we estimate how many mutations have occurred between two viral genomes exhibiting a range of alignment scores, which can provide a general estimate of the genomic difference between two sequences deemed similar via a specific threshold. This is not a part of our general graph-examination approach (Section 8.2), but might be used to justify or fine-tune threshold choices in future work.

We randomly select one sequence as a 'start sequence' from our data set. Beginning with each start sequence, a random single base change (SBC) is introduced, creating a second sequence differing from the first by only one base. A random single base change is then introduced to the second sequence, creating a third, and so on. 200 sequences are created in this manner and all pairwise ClustalW alignment scores are calculated for each set. Tables 8.13- 8.20 show the average number of mutations between sequences exhibiting a range of ClustalW alignment scores.

Table 8.13: Average number of single base changes for a range of ClustalW Alignment scores, segment 1

ClustalW alignment score (%)	Ave # SBC's	# samples
100	1.62	218
99	19.42	6654
98	55.84	5282
97	92.87	4069
96	130.40	2676
95	167.75	1181
94	196.60	20

Table 8.14: Average number of single base changes for a range of ClustalW alignment scores, segment 2

ClustalW alignment score (%)	Ave # SBCs	# samples
100	1.61	216
99	19.35	6645
98	55.97	5336
97	92.88	3998
96	128.84	2517
95	163.67	1279
94	191.06	109

Table 8.15: Average number of single base changes for a range of ClustalW alignment scores, segment 3

ClustalW alignment score (%)	Ave # SBCs	# samples
100	1.48	199
99	18.47	6320
98	52.62	5073
97	86.81	3898
96	122.70	2957
95	158.45	1425
94	186.20	228

Table 8.16: Average number of single base changes for a range of ClustalW alignment scores, segment 4

ClustalW alignment score (%)	Ave # SBCs	# samples
100	1.67	248
99	14.95	5002
98	41.55	4175
97	67.01	3356
96	92.38	2822
95	118.90	2206
94	145.30	1432
93	170.03	719
92	189.59	140

Table 8.17: Average number of single base changes for a range of ClustalW alignment scores, segment 5

ClustalW alignment score (%)	Ave # SBCs	# samples
100	1.48	199
99	13.44	4508
98	36.65	3723
97	59.76	3382
96	83.56	2733
95	106.69	2190
94	130.27	1694
93	153.62	1062
92	175.93	555
91	193.89	54

Table 8.18: Average number of single base changes for a range of ClustalW alignment scores, segment 6

ClustalW alignment score (%)	Ave # SBCs	# samples
100	1.67	248
99	12.90	4246
98	35.02	3621
97	55.93	2916
96	76.04	2473
95	96.85	2306
94	118.72	1786
93	140.11	1279
92	160.91	812
91	180.74	387
90	196.15	26

Table 8.19: Average number of single base changes for a range of ClustalW alignment scores, segment 7

ClustalW alignment score (%)	Ave # SBCs	# samples
100	1.68	250
99	9.07	2819
98	24.16	2733
97	39.87	2555
96	55.08	2189
95	70.41	1965
94	85.45	1855
93	101.53	1622
92	118.80	1468
91	137.42	1210
90	156.04	824
89	172.50	404
88	186.65	2

Table 8.20: Average number of single base changes for a range of ClustalW alignment scores, segment 8

ClustalW alignment score (%)	Ave # SBCs	# samples
100	1.67	248
99	8.29	2530
98	21.47	2362
97	34.33	2109
96	48.02	2236
95	61.61	1805
94	75.88	1976
93	91.15	1540
92	105.40	1448
91	119.89	1145
90	135.01	1037
89	149.89	713
88	165.74	610
87	182.08	337

8.5 Summary

In this chapter, we describe a method for generating a network from viral genomes. We also apply this method to an *Influenza A Virus* dataset and examine the patterns of connectivity between sequence groups.

In the network generated here, the non-human host type showing the highest number of edges with Human vertices is Swine. Edges are found between Human and Swine vertices from different and same countries: New Zealand Human/Canada Swine, New Zealand Human/China Swine, and Canada Human/Canada Swine. A lesser number of edges are found between Human and Avian vertex types. The Avian types are primarily Domestic. All of these edges occur within the same country of origin and are found within Canada, the Netherlands, and Thailand. Segments 5 and 8 contribute the most the Human/Swine and Human/Avian edges. In contrast, segments 7 and 8 contribute the most to edges found between Domestic and Wild Avian vertices.

Examining inter-year edges shows that the majority are found between consecutive years. Additionally, segment 8 contributes to 70% of these edges.

The analysis we describe here is not all inclusive in that it might be adapted to examine several other aspects of viral transmission with different thresholds applied. This type of analysis might be improved by allowing more accurate comparisons by making the distribution of vertex groups more uniform. However, we feel that this method is promising and can provide insight into some aspects of global and local disease transmission.

Chapter 9

Conclusions and Future Work

The research described in this dissertation is focused on two major components. The first is approximating complete, whole-genome alignment scores with a k-mer based, alignment-free method. We examine existing word-based methods, and develop two of our own. Each method is tested for accuracy against the popular ClustalW pairwise alignment scores, and we select one such metric which we believe is the most accurate.

The second introduces a method for developing a graph to model and examine similarity relationships between whole genome sequences. The method uses a k-mer alignment-free distance metric to efficiently compute pairwise distances of all sequences of interest, and draws on existing tools to examine characteristics of graphs. We present a detailed example of this method applied to a complex *Influenza A virus* dataset, and discover some interesting results, which are discussed in detail.

Table 9.1 provides a summary of the benefits of our approach over methods traditionally used in similar studies. These benefits include time- and space- efficient computations providing the ability to study large datasets. Strengths of our approach also include a general graph formation, as opposed to phylogenetic tree construction, without restrictions on node degree. Furthermore, graph theory is a well developed field of study with many tools to draw from for designing further analysis.

Although our research is presented mostly in conjunction with viral sequences, all components of the tools and methods developed here may be applicable to whole genome sequences that are somewhat similar and show relatively fast mutation rates. This makes our method applicable to a variety of datasets and enables researchers to

Table 9.1: Benefits of the described methods

	Method Described	Standard Approach
Sequence Comparisons	Sequence comparisons via fast and accurate estimates of alignment scores allow for large datasets of whole-genomic sequences to be studied.	Standard pairwise alignments are computationally intensive, which may restrict the number of sequences and/or sequence lengths to be included in a study.
Graph-Based Analysis	A graph allows for flexibility in the placement of edges and node degree. Existing graph theoretic approaches may also be applied, such as mixing patterns.	Traditional phylogenetic tree approaches used to estimate evolutionary distances between elements are computationally intensive and typically are based on node degree restrictions.

examine a number of geographic trends of genomic variability.

A potential application of our approach could be to determine infection sources in disease outbreaks, particularly in cases where timely results are desired. In such outbreaks, sequencing of large numbers of viral or bacterial samples could provide large sequence sets which would be efficiently analyzed using a linear-time comparison method and graph-based approach described. In the event of an outbreak, it would be possible to collect and sequence viral samples from infected people from different geographic locations. For a set of sequences similar in size to the *Influenza A Virus* dataset used in Chapter 8, the time to build transmission graphs using the described method on one dual core computer with 8GB of RAM would be less than one day. Comparing all sequences and generating a phylogenetic tree with ClustalW would require about six days. Mixing patterns could indicate the most probable routes of transmission. If viral isolates were collected at different time intervals, mixing pattern analysis in conjunction with sample collection times could indicate the direction of transmission as well. For example, if several vertices showed a high probability of forming edges with vertices located west of them and collected at an earlier date, the

indication would be that the disease was spreading in an eastward fashion.

Other applications may include generating a graph to represent similarities between human genomic sequences collected in distributed locations. This could yield insight into the geographic aspect of disease susceptibility in human populations. Our method could locate highly connected vertices, or clusters, of similar genotypes which overlay geographically with disease instances. The probability of disease transmission between certain genotypic types may also be indicated by mixing patterns.

For ecological-based studies, comparisons of viral samples in non-human, animal populations such as rodents or migratory avian species may prove useful when studying the movement of these species groups. As it can be assumed that, in general, viral transmission requires physical proximity, geographic routes of viral transmission would indicate the physical movement of animals carrying the virus. Mixing pattern analysis could indicate highly utilized routes of movement or the interaction of certain species groups.

The work presented here might be expanded in several interesting directions. The first is the further development of an alignment-free metric to approximate dynamic programming alignment scores in a more general setting. The research here focuses on small, viral genomes. Our methods are not tested on a diverse set of longer genomes, and so our results cannot be easily extrapolated to general instances. In addition, the method we found most useful (**KMOD**) was developed experimentally and has not been justified mathematically.

As a second possible expansion of this study, we would like to consider generating graphs that are based on directed, rather than undirected edges to represent transmission from isolates collected at an earlier date to those collected at a later date. Using appropriate tools from graph theory to study directed graphs, we might be able to study transmission patterns throughout time.

A third additional investigation can include the study of more complex connectivity properties of graphs, and applications of these to our disease network models. Studying graph characteristics such as connected components and cliques might en-

able us to examine connectivity patterns across and within certain vertex groups.

A fourth possible direction for future work is to develop a normalization procedure of viral datasets so that resulting graphs can be used to more accurately extrapolate general transmission tendencies around the globe. The *Influenza A virus* dataset is highly skewed with regard to many characteristics. For example, approximately half of all sequences originate from the United States. The second most highly represented country is New Zealand, contributing approximately 13% of all samples. Human samples contribute to 65% of all sequences. Future work should investigate the underlying network structure of influenza in more evenly distributed data sets.

Future work will also include making the C++ code used for k-mer comparisons and graph creation publicly accessible.

Bibliography

[1] D.G. Altman and J.M. Bland. Diagnostic tests. 1: Sensitivity and specificity. *BMJ: British Medical Journal*, 308(6943):1552, 1994.

[2] Y. Bao, P. Bolotov, D. Dernovoy, B. Kiryutin, L. Zaslavsky, T. Tatusova, J. Ostell, and D. Lipman. The influenza virus resource at the National Center for Biotechnology Information. *Journal of virology*, 82(2):596, 2008.

[3] J. Bohlin and E. Skjerve. Examination of Genome Homogeneity in Prokaryotes Using Genomic Signatures. *PLoS One*, 4(12):e8113, 2009.

[4] J. Bohlin, E. Skjerve, and D.W. Ussery. Investigations of oligonucleotide usage variance within and between prokaryotes. *PLoS Computational Biology*, 4(4), 2008.

[5] M. Bonet, M. Steel, T. Warnow, and S. Yooseph. Better methods for solving parsimony and compatibility. *Journal of Computational Biology*, 5(3):391–407, 1998.

[6] A. Breland, S. Nasser, K. Schlauch, M. Nicolescu, and F.C. Harris Jr. Efficient Influenza A Virus Origin Detection. *Journal of Electronics & Computer Science*, 10(2), 2008.

[7] A. Breland, K. Schlauch, M. Nicolescu, and F.C. Harris Jr. An annotated k-deep prefix tree for (1-k)-mer based sequence comparisons. In *Proceedings of the First ACM International Conference on Bioinformatics and Computational Biology*, pages 79–85. ACM, 2010.

[8] L. Brocchieri. Phylogenetic inferences from molecular sequences: review and critique. *Theoretical Population Biology*, 59(1):27–40, 2001.

[9] P. Chain, S. Kurtz, E. Ohlebusch, and T. Slezak. An applications-focused review of comparative genomics tools: Capabilities, limitations and future challenges. *Briefings in Bioinformatics*, 4(2):105, 2003.

[10] J.P. Chretien, J.C. Gaydos, J.L. Malone, and D.L. Blazes. Global network could avert pandemics. *Nature*, 440(7080):25–26, Mar 2006.

[11] T. De Oliveira, O.G. Pybus, A. Rambaut, M. Salemi, S. Cassol, M. Ciccozzi, G. Rezza, G.C. Gattinara, R. D'Arrigo, M. Amicosante, L. Perrin, V. Colizzi, C. Perno, and Benghazi Study Group. Molecular epidemiology: HIV-1 and HCV sequences from Libyan outbreak. *Nature*, 444(7121):836–837, 2006.

[12] A.L. Delcher, S. Kasif, R.D. Fleischmann, J. Peterson, O. White, and S.L. Salzberg. Alignment of whole genomes. *Nucleic Acids Research*, 27(11):2369, 1999.

[13] A.L. Delcher, A. Phillippy, J. Carlton, and S.L. Salzberg. Fast algorithms for large-scale genome alignment and comparison. *Nucleic Acids Research*, 30(11):2478, 2002.

[14] F. Delsuc, H. Brinkmann, and H. Philippe. Phylogenomics and the reconstruction of the tree of life. *Nature Reviews Genetics*, 6(5):361–375, 2005.

[15] J.S. Deogun, F. Ma, J. Yang, and A. Benson. A prototype for multiple whole genome alignment. *Hawaii International Conference on System Sciences*, 9:275b, 2003.

[16] T. Dobzhansky. Nothing in biology makes sense except in the light of evolution. *The American Biology Teacher*, 35(3):125–129, 1973.

[17] W.F. Doolittle. Phylogenetic classification and the universal tree. *Science*, 284(5423):2124, 1999.

[18] R.C. Edgar. Local homology recognition and distance measures in linear time using compressed amino acid alphabets. *Nucleic acids research*, 32(1):380, 2004.

[19] R.C. Edgar and S. Batzoglou. Multiple sequence alignment. *Current Opinion in Structural Biology*, 16(3):368–373, 2006.

[20] J.A. Eisen and C.M. Fraser. Phylogenomics: intersection of evolution and genomics. *Science*, 300(5626):1706, 2003.

[21] I. Elias. Settling the intractability of multiple alignment. *Journal of Computational Biology*, 13(7):1323–1339, 2006.

[22] T.D. Emmanouil, A. Reymond, N. Scamuffa, C. Ucla, E. Kirkness, C. Rossier, and S.E. Antonarakis. Evolutionary discrimination of mammalian conserved non-genic sequences (cngs). *Science*, 302:1033 – 1035, 2003.

[23] T. Fawcett. An introduction to ROC analysis. *Pattern recognition letters*, 27(8):861–874, 2006.

[24] Y. Fofanov, Y. Luo, C. Katili, J. Wang, Y. Belosludtsev, T. Powdrill, C. Belapurkar, V. Fofanov, T.B. Li, S. Chumakov, and M. Pettitt. How independent are the appearances of n-mers in different genomes? *Bioinformatics*, 20(15):2421, 2004.

[25] C.M. Fraser-Liggett. Insights on biology and evolution from microbial genome sequencing. *Genome research*, 15(12):1603, 2005.

[26] G. Ganapathy, V. Ramachandran, and T. Warnow. Better hill-climbing searches for parsimony. *Algorithms in Bioinformatics*, pages 245–258, 2003.

[27] R.L. Graham and L.R. Foulds. Unlikelihood that minimal phylogenies for a realistic biological study can be constructed in reasonable computational time. *Mathematical Biosciences*, 60(2):133–142, 1982.

[28] G. Hampikian and T. Andersen. Absent sequences: nullomers and primes. In *Pacific Symposium on Biocomputing 2007: Maui, Hawaii, 3-7 January 2007*, page 355. World Scientific Pub Co Inc, 2006.

[29] S. Hazelhurst. An efficient implementation of the d 2 distance function for EST clustering: preliminary investigations. In *Proceedings of the 2004 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries*, pages 229–233. South African Institute for Computer Scientists and Information Technologists, 2004.

[30] E.C. Holmes. RNA virus genomics: a world of possibilities. *The Journal of clinical investigation*, 119(9):2488, 2009.

[31] C.H. House and S.T. Fitz-Gibbon. Using homolog groups to create a whole-genomic tree of free-living organisms: an update. *Journal of molecular evolution*, 54(4):539–547, 2002.

[32] G.M. Jenkins, A. Rambaut, O.G. Pybus, and E.C. Holmes. Rates of molecular evolution in RNA viruses: a quantitative phylogenetic analysis. *Journal of molecular evolution*, 54(2):156–165, 2002.

[33] S.R. Jun, G.E. Sims, G.A. Wu, and S.H. Kim. Whole-proteome phylogeny of prokaryotes by feature frequency profiles: An alignment-free method with optimal feature resolution. *Proceedings of the National Academy of Sciences*, 107(1):133, 2010.

[34] S. Karlin. Global dinucleotide signatures and analysis of genomic heterogeneity. *Current opinion in microbiology*, 1(5):598–610, 1998.

[35] S. Karlin and C. Burge. Dinucleotide relative abundance extremes: a genomic signature. *Trends in genetics*, 11(7):283–290, 1995.

[36] S. Karlin, A.M. Campbell, and J. Mrazek. Comparative DNA analysis across diverse genomes. *Annual Review of Genetics*, 32(1):185–225, 1998.

[37] M.J. Keeling and K.T.D. Eames. Networks and epidemic models. *J R Soc Interface*, 2(4):295–307, Sep 2005.

[38] B. Kolaczkowski and J.W. Thornton. Performance of maximum parsimony and likelihood phylogenetics when evolution is heterogeneous. *Nature*, 431(7011):980–984, 2004.

[39] S. Kurtz, A. Phillippy, A. Delcher, M. Smoot, M. Shumway, C. Antonescu, and S. Salzberg. Versatile and open software for comparing large genomes. *Genome biology*, 5(2):R12, 2004.

[40] M.A. Larkin, G. Blackshields, N.P. Brown, R. Chenna, P.A. McGettigan, H. McWilliam, F. Valentin, I.M. Wallace, A. Wilm, R. Lopez, J.D. Thompson, T.J. Gibson, and D.G. Higgins. Clustal W and Clustal X version 2.0. *Bioinformatics*, 23(21):2947, 2007.

[41] T. Leitner, B. Foley, B. Hahn, P. Marx, F. McCutchan, J. Mellors, S. Wolinsky, and B. Korber. HIV sequence compendium 2005. *Theoretical Biology and Biophysics Group, Los Alamos National Laboratory, Los Alamos, NM*, 2005.

[42] P. Lemey, A. Rambaut, A.J. Drummond, and M.A. Suchard. Bayesian phylogeography finds its roots. *PLoS computational biology*, 5(9):e1000520, 2009.

[43] M.Y. Leung, G.M. Marsh, and T.P. Speed. Over-and underrepresentation of short DNA words in herpesvirus genomes. *Journal of Computational Biology*, 3(3):345–360, 1996.

[44] Q. Li, Z. Xu, and B. Hao. Composition vector approach to whole-genome-based prokaryotic phylogeny: Success and foundations. *Journal of Biotechnology*, 2009.

[45] W. Li, D. Raoult, and P.E. Fournier. Bacterial strain typing in the genomic era. *FEMS microbiology reviews*, 33(5):892–916, 2009.

[46] LLC Los Alamos National Security. Hiv databases. `http://www.hiv.lanl.gov`.

[47] G. Lunter, A. Rocco, N. Mimouni, A. Heger, A. Caldeira, and J. Hein. Uncertainty in homology inferences: assessing and improving genomic sequence alignment. *Genome Research*, 18(2):298, 2008.

[48] K.D. Makova, S. Yang, and F. Chiaromonte. Insertions and deletions are male biased too: a whole-genome analysis in rodents. *Genome research*, 14(4):567, 2004.

[49] E.H. Margulies. Confidence in comparative genomics. *Genome research*, 18(2):199, 2008.

[50] E.H. Margulies and E. Birney. Approaches to comparative sequence analysis: towards a functional view of vertebrate genomes. *Nature Reviews Genetics*, 9(4):303–313, 2008.

[51] M. Margulies, M. Egholm, W.E. Altman, S. Attiya, J.S. Bader, L.A. Bemben, J. Berka, M.S. Braverman, Y Chen, Z Chen, S.B. Dewell, L. Du, J.M. Fierro, X.V. Gomes, B.C. Goodwin, W. He, S. Helgesen, C. Ho, G.P. Irzyk, S.C. Jando, M.L.I. Alenquer, T.P. Jarvie, K.B. Jirage, J. Kim, J.R Knight, J.R. Lanza, J.H. Leamon, S.M. Lefkowitz, M. Lei, J. Li, K.L. Lohman, H. Lu, V.B. Makhijani, K.E. McDade, M.P. McKenna, E.W. Myers, E. Nickerson, J.R. Nobile, R. Plant, B.P. Puc, M.T. Ronan, G.T. Roth, G.J. Sarkis, J.F. Simons, J.W. Simpson, M. Srinivasan, K.R. Tartaro, A. Tomasz, K.A. Vogt, G.A. Volkmer, S.H. Wang, Y. Wang, M.P. Weiner, P. Yu, R.F. Begley, and J.M. Rothberg. Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, 437(7057):376–380, 2005.

[52] R. Merkl, M. Kroger, P. Rice, and H.J. Fritz. Statistical evaluation and biological interpretation of non-random abundance in the E. coli K-12 genome of tetra-and pentanucleotide sequences related to VSP DNA mismatch repair. *Nucleic acids research*, 20(7):1657, 1992.

[53] M.L. Metzker. Sequencing technologiesthe next generation. *Nature Reviews Genetics*, 11(1):31–46, 2009.

[54] W. Miller, K.D. Makova, A. Nekrutenko, and R.C. Hardison. Comparative genomics. *Annual Review of Genomics and Human Genetics*, 2004.

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

[56] M.I. Nelson, L. Edelman, D.J. Spiro, A.R. Boyne, J. Bera, R. Halpin, N. Sengamalay, E. Ghedin, M.A. Miller, L. Simonsen, C. Viboud, and E.C. Holmes. Molecular epidemiology of a/h3n2 and a/h1n1 influenza virus during a single epidemic season in the united states. *PLoS Pathog*, 4(8):e1000133, Aug 2008.

[57] M.I. Nelson, L. Simonsen, C. Viboud, M.A. Miller, and E.C. Holmes. Phylogenetic analysis reveals the global migration of seasonal influenza A viruses. *PLoS Pathog*, 3(9):1220–1228, 2007.

[58] M.E.J. Newman. The structure and function of complex networks. *SIAM review*, 45(2):167–256, 2003.

[59] C. Notredame. Recent progress in multiple sequence alignment: a survey. *pgs*, 3(1):131–144, 2002.

[60] G.J. Olsen, C.R. Woese, and R. Overbeek. The winds of (evolutionary) change: breathing new life into microbiology. *Journal of Bacteriology*, 176(1):1, 1994.

[61] Ou, C-Y., C.A. Ciesielski, G. Myers, C.I. Bandea, C-C. Luo, B.T.M. Korber, J.I. Mullins, G. Schochetman, R.L. Berkelman, A.N. Economou, J.J. Witte, L.J Furman, G.A. Satten, K.A. Maclnnes, J.W. Curran, H.W. Jaffe, Laboratory Investigation Group, and Epidemiologic Investigation Group. Molecular epidemiology of HIV transmission in a dental practice. *Science*, 256(5060):1165, 1992.

[62] J. Parkhill, B.W. Wren, N.R. Thomson, R.W. Titball, M.T.G. Holden, M.B. Prentice, M. Sebaihia, K.D. James, C. Churcher, K.L. Mungall, S. Baker, D. Basham, S.D. Bentley, K. Brooks, A.M. Cerdeo-Trraga, T. Chillingworth, A. Cronin, R.M. Davies, P. Davis, G. Dougan, T. Feltwell, N. Hamlin, S. Holroyd, K. Jagels, A.V. Karlyshev, S. Leather, S. Moule, P.C.F. Oyston, M. Quail, K. Rutherford, M. Simmonds, J. Skelton, K. Stevens, S. Whitehead, and B.G. Barrell. Genome sequence of Yersinia pestis, the causative agent of plague. *Nature*, 413(6855):523–527, 2001.

[63] PatentLens. The influenza genome comprises eight segments. `http://www.patentlens.net/daisy/influenza/4133/3929.html`.

[64] H. Philippe, F. Delsuc, H. Brinkmann, and N. Lartillot. Phylogenomics. *Annual Review of Ecology, Evolution, and Systematics*, 36:541–562, 2005.

[65] D.T. Pride, R.J. Meinersmann, T.M. Wassenaar, and M.J. Blaser. Evolutionary implications of microbial genome tetranucleotide frequency biases. *Genome research*, 13(2):145, 2003.

[66] J. Qi, B. Wang, and B.I. Hao. Whole proteome prokaryote phylogeny without sequence alignment: a K-string composition approach. *Journal of Molecular Evolution*, 58(1):1–11, 2004.

[67] A. Rambaut, O.G. Pybus, M.I. Nelson, C. Viboud, J.K. Taubenberger, and E.C. Holmes. The genomic and epidemiological dynamics of human influenza A virus. *Nature*, 453(7195):615–619, 2008.

[68] B.D. Redelings and M.A. Suchard. Incorporating indel information into phylogeny estimation for rapidly emerging pathogens. *BMC Evolutionary Biology*, 7(1):40, 2007.

[69] W. Resch, L. Zaslavsky, B. Kiryutin, M. Rozanov, Y. Bao, and T.A. Tatusova. Virus variation resources at the National Center for Biotechnology Information: dengue virus. *BMC microbiology*, 9(1):65, 2009.

[70] S. Schbath, B. Prum, and E. De Turckheim. Exceptional motifs in different Markov chain models for a statistical analysis of DNA sequences. *Journal of Computational Biology*, 2(3):417–437, 1995.

[71] D. Secko. A monks flourishing garden: the basics of molecular biology explained, 2003.

[72] L. Simonsen, C. Viboud, B.T. Grenfell, J. Dushoff, L. Jennings, M. Smit, C. Macken, M. Hata, J. Gog, M.A. Miller, and E.C. Holmes. The genesis and spread of reassortment human influenza A/H3N2 viruses conferring adamantane resistance. *Molecular biology and evolution*, 24(8):1811, 2007.

[73] G.E. Sims, S.R. Jun, G.A. Wu, and S.H. Kim. Alignment-free genome comparison with feature frequency profiles (FFP) and optimal resolutions. *Proceedings of the National Academy of Sciences*, 106(8):2677, 2009.

[74] T.F. Smith and M.S. Waterman. Identification of common molecular subsequences. *Journal of molecular biology*, 147(1):195–197, 1981.

[75] B. Snel, M.A. Huynen, and B.E. Dutilh. Genome trees and the nature of genome evolution. *Annu. Rev. Microbiol.*, 59:191–209, 2005.

[76] W. Stallings. *Computer organization and architecture: designing for performance*. Prentice Hall, 2009.

[77] G. Stuart, K. Moffett, and R.F. Bozarth. A whole genome perspective on the phylogeny of the plant virus family Tombusviridae. *Archives of virology*, 149(8):1595–1610, 2004.

[78] C.J. Stubben, M.L. Duffield, I.A. Cooper, D.C. Ford, J.D. Gans, A.V. Karlyshev, B. Lingard, P.C.F. Oyston, A. De Rochefort, J. Song, B. Wren, R. Titball, and M. Wolinsky. Steps toward broad-spectrum therapeutics: discovering virulence-associated genes present in diverse human pathogens. *BMC genomics*, 10(1):501, 2009.

[79] Y. Sun, Y. Cai, L. Liu, F. Yu, M.L. Farrell, W. McKendree, and W. Farmerie. ESPRIT: estimating species richness using large collections of 16S rRNA pyrosequences. *Nucleic acids research*, 37(10):e76, 2009.

[80] H. Tettelin, N.J. Saunders, J. Heidelberg, A.C. Jeffries, K.E. Nelson, J.A. Eisen, K.A. Ketchum, D.W. Hood, J.F. Peden, R.J. Dodson, W.C. Nelson, M.L. Gwinn, R. DeBoy, J.D. Peterson, E.K. Hickey, D.H. Haft, S.L. Salzberg, O. White, R.D. Fleischmann, B.A. Dougherty, T Mason, A. Ciecko, D.S. Parksey, E. Blair, H. Cittone, E.B. Clark, M.D. Cotton, T.R. Utterback, H. Khouri, H. Qin, J. Vamathevan, J. Gill, V. Scarlato, V. Masignani, M. Pizza, G. Grandi, L. Sun, H.O. Smith, C.M. Fraser, E.R. Moxon, R. Rappuoli, and J.C. Venter. Complete genome sequence of Neisseria meningitidis serogroup B strain MC58. *Science*, 287(5459):1809, 2000.

[81] D.C. Torney, C. Burks, D. Davison, and K.M. Sirotkin. *Computers and DNA, SFI Studies in the Sciences of Complexity*. Addison-Wesley Publishing Co., 1990.

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

[83] L. Wang and T. Jiang. On the complexity of multiple sequence alignment. *Journal of computational biology*, 1(4):337–348, 1994.

[84] H.T. Wareham. A simplified proof of the NP-and MAX SNP-hardness of multiple sequence tree alignment. *Journal of Computational Biology*, 2(4):509–514, 1995.

[85] R.L. Warren, G.G. Sutton, S.J.M. Jones, and R.A. Holt. Assembling millions of short DNA sequences using SSAKE. *Bioinformatics*, 23(4):500, 2007.

[86] R.G. Webster. The importance of animal influenza for human disease. *Vaccine*, 20:S16–S20, 2002.

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

[88] Y.I. Wolf, I.B. Rogozin, N.V. Grishin, and E.V. Koonin. Genome trees and the tree of life. *TRENDS in Genetics*, 18(9):472–479, 2002.

[89] K.M. Wong, M.A. Suchard, and J.P. Huelsenbeck. Alignment uncertainty and genomic analysis. *Science*, 319(5862):416–417, 2008.

[90] G.A. Wu, S.R. Jun, G.E. Sims, and S.H. Kim. Whole-proteome phylogeny of large dsDNA virus families by an alignment-free method. *Proceedings of the National Academy of Sciences*, 106(31):12826, 2009.

[91] A.C. Yang, A.L. Goldberger, and C.K. Peng. Genomic classification using an information-based similarity index: application to the SARS coronavirus. *Journal of Computational Biology*, 12(8):1103–1116, 2005.

[92] E. Zuckerkandl and L. Pauling. Molecules as documents of evolutionary history. *Journal of Theoretical Biology*, 8(2):357–366, 1965.