10-9-2018 10:30 AM

DNA Sequence Classification: It's Easier Than You Think: An open-source k-mer based machine learning tool for fast and accurate classification of a variety of genomic datasets

Stephen Solis-Reyes
The University of Western Ontario

Supervisor
Kari, Lila
The University of Western Ontario

Graduate Program in Computer Science
A thesis submitted in partial fulfillment of the requirements for the degree in Master of Science

Abstract

Supervised classification of genomic sequences is a challenging, well-studied problem with a variety of important applications. We propose an open-source, supervised, alignment-free, highly general method for sequence classification that operates on k-mer proportions of DNA sequences. This method was implemented in a fully standalone general-purpose software package called KAMERIS, publicly available under a permissive open-source license. Compared to competing software, ours provides key advantages in terms of data security and privacy, transparency, and reproducibility. We perform a detailed study of its accuracy and performance on a wide variety of classification tasks, including virus subtyping, taxonomic classification, and human haplogroup assignment. We demonstrate the success of our method on whole mitochondrial, nuclear, plastid, plasmid, and viral genomes, as well as randomly sampled eukaryote genomes and transcriptomes. Further, we perform head-to-head evaluations on the tasks of HIV-1 virus subtyping and bacterial taxonomic classification with a number of competing state-of-the-art software solutions, and show that we match or exceed all other tested software in terms of accuracy and speed.

Keywords: sequence classification; machine learning; alignment-free; k-mers; virus subtyping; comparative genomics; open-source

Co-Authorship Statement

Chapter 3 of this thesis contains a version of the article "An open-source k-mer based machine learning tool for fast and accurate subtyping of HIV-1 genomes" by Stephen Solis-Reyes, Mariano Avino, Art Poon, and Lila Kari. It has been accepted for publication in the journal PLoS One. The author order follows the conventions of the field: it has two senior authors (LK, AP) and the author order reflects the contributions of the authors. The individual contributions are as follows. SSR: data collection; data analysis; methodology and result interpretation; manuscript draft; manuscript editing; software development. MA: data analysis; methodology and result interpretation; manuscript editing. AP: data analysis; methodology and result interpretation; manuscript draft; manuscript editing. LK: data analysis; methodology and result interpretation; manuscript draft; manuscript editing.

Acknowledgments

First, I thank my supervisor, Dr. Lila Kari for her support and guidance – without her, this thesis would not exist.

Also, I thank my lab-mates Dr. Rallis Karamichalis, Gurjit Randhawa, and Zhihao Wang for a wide variety of helpful discussions and insights, and I thank Dr. Art Poon, Dr. Mariano Avino, and Dr. David Smith for their wealth of biological expertise.

Finally, I thank my parents for their constant encouragement and support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The *sequence classification problem* may be stated as follows: given a set of genomic sequences (in this work, DNA or RNA sequences) partitioned into some known groups, and a sequence not in the known set, predict which group the new sequence belongs to. This is an important problem in the field of bioinformatics because several well-studied, more specific problems are instances of this one: for example, the *virus subtyping problem*, where we wish to assign a viral sequence to its subtype, or the *taxonomic classification problem*, where we wish to determine the phylogenetic group of an organism given some of its genomic sequence data, or the *haplogroup identification problem*, where we assign a human mitochondrial sequence to its haplogroup, allowing the identification of its maternal lineage.

A tremendous variety of methods have been applied to this problem, including both alignment-based and alignment-free methods. Our goal is to develop an even better, more efficient, more accurate method than the state-of-the-art, which we achieve by proposing, in this work, a remarkably simple but extremely general method. It works by first taking a DNA sequence and computing a vector of the proportions of every possible k-mer (that is, every

length-k substring). These vectors are used as feature vectors, and well-known supervised classification algorithms are trained on the vectors.

We develop an open-source, easy-to-use, standalone software implementation of our method, which we call KAMERIS, available at https://github.com/stephensolis/kameris, including easy-to-follow setup and use instructions. As a standalone application, we avoid the need for researchers to transmit sequence data to a remote server, eliminating privacy and security concerns. Further, as an open-source application, researchers have full visibility into the implementation of the algorithm, and can reproduce results at any time with a copy of a previous version of the software, which is not possible with an opaque server-based solution.

One goal of this work was straightforward reproducibility of results, and to that end, every experiment presented here can be easily reproduced by following the step-by-step instructions at https://github.com/stephensolis/kameris-experiments. On the same page, every sequence and its metadata from every dataset referenced here is available as well, to aid in future work building on our results.

We curate and use a large variety of datasets to validate the performance of our method, and we compute its accuracy on a variety of tasks. In Chapter (which is a version of our paper "An open-source k-mer based machine learning tool for fast and accurate subtyping of HIV-1 genomes", accepted for publication in PLoS One), we focus on the virus subtyping problem, demonstrating our performance on the classification of HIV-1, dengue, influenza A, hepatitis B, and hepatitis C virus genomes; in addition to working with whole genomes, we demonstrate we maintain high accuracy even when working with just the HIV-1 *pol* gene and also with randomly sampled HIV-1 genomes. Further, we perform a head-to-head comparison with four competing state-of-the-

art HIV-1 subtyping tools, namely CASTOR, COMET, SCUEAL, and REGA, and show that we match or exceed all in terms of accuracy and speed. In Chapter 4, we go on to consider the taxonomic classification of whole mitochondrial, nuclear, plastid, plasmid, and viral genomes, and randomly sampled marine eukaryote and plant transcriptomes, into taxonomic groupings at every level from kingdom down to genus; and the determination of human haplogroups from whole mitochondrial genomes. We again perform a head-to-head comparison with a competing tool called LAF on the taxonomic classification of whole bacterial genomes, and demonstrate a higher classification accuracy. In total, across all datasets and experiments, we use about 470,000 unique sequences comprising a total length of over 276 Gbp of sequence data.

We conclude by discussing possible extensions to the current work, including a method for identifying particularly important k-mers from the perspective of classification, an application to the detection of 'mixed' or chimeric sequences, and even more challenging tasks such as the classification of unfiltered next-generation sequencing (NGS) read data and the diagnosis of genetic disease.

Chapter 2

Literature review

2.1 Biological background

Earth has a great diversity of living organisms. For thousands of years, people have sought to categorize these organisms and explore the relationships between them. In modern use, from most broad to most specific, organisms are principally organized into the ranks of kingdom, phylum or division, class, order, family, genus, and species. Species can be further subdivided – for instance, humans are divided by common ancestry into haplogroups.

In the days before genomics and molecular biology, scientists took a morphological approach, performing categorization by comparing the form and structural features of organisms. Nowadays, however, scientists use information from the DNA of organisms to do this, and many methods have been proposed for doing so, ranging from DNA barcoding [64], to sequence alignment, to a wide variety of alignment-free methods.

DNA (deoxyribonucleic acid) is a long stranded molecule which can be viewed as a string on a four letter alphabet: A, C, G, T, where each letter

represents one of the four basic constituent molecules collectively known as nucleotides: cytosine (C), guanine (G), adenine (A), and thymine (T). DNA may exist in either a single-stranded or double-stranded form – when double-stranded, each type of nucleotide binds with its complementary pair on the opposite strand: A pairs with T and C with G. As well, the two strands of DNA run in opposite directions to each other, so the sequences of each strand are reverse complements of each other. DNA may be present in several parts of a cell. For cells with a nucleus or nucleoid, the majority of its genome is located there. Most eukaryotes have further genetic material in their mitochondria, as do plants, algae, and some other eukaryotes in their plastids, and bacteria in plasmid molecules. In Section 4.1, we explore sequence classification using genetic information from each of these regions.

Within an organism, DNA encodes information on how to assemble proteins. Proteins, large molecules consisting of long chains of amino acids, are an essential part of every organism and are responsible for a wide range of functions within a cell. The central dogma of molecular biology, first described by Crick [29], describes how this encoding works: first, DNA is processed and spliced into mRNA (messenger RNA) molecules in a process called transcription; then, mRNA is translated into proteins by the ribosome. Sequences of DNA or RNA which code for a specific protein are known as genes, and the specific portion of the gene directly processed by the ribosome is called the coding region or CDS (coding sequence) of the gene. The set of all DNA in an organism is called its genome, the set of all RNA produced during transcription is called the transcriptome, and the set of all coding regions from all genes is called the exome. In Section 4.1, we explore sequence classification using transcriptome and exome data, and how its performance compares with genome data.

In order to use DNA for sequence classification, it is first necessary to read or sequence it, that is, to determine the order of nucleotides on the DNA strand. Generally, it is not known how to read more than a few tens of thousands of nucleotides at a time, so in order to sequence a long DNA molecule (such as a chromosome), it is necessary to first break the DNA up into small pieces, read each piece independently, and then use software to stitch the pieces together based on overlapping fragments in a process known as sequence assembly. Depending on the sequencing technology being used, it may be possible to read longer or shorter fragments; so-called next-generation or high-throughput sequencing methods are able to lower sequencing costs and increase sequencing speed by reading shorter fragments, usually of a few hundred nucleotides, and parallelizing the sequencing process, producing thousands or millions of sequences concurrently. However, shorter, more numerous sequence fragments make sequence assembly more computationally expensive due to the larger amount of data to be processed. For this reason, it is useful to have sequence classification methods which do not depend on alignment, and we explore this use-case in Sections 3.4 and 4.3.

2.2 Alignment-based methods

Alignment-based methods, broadly, are any methods based on searching for base-to-base correspondences in two or more sequences [182]. These methods measure sequence similarity by computing a score based on the number of matches and mismatches between sequences – in this way, they may compute the class of a given query sequence by locating the most similar sequence in the known set. There have been many alignment-based tools developed, including single-sequence aligners such as BLAST [6] and FASTA [102],

and multiple-sequence aligners such as ClustalW [159] and MUSCLE [42].

Since alignment-based tools can report the exact regions of high similarity between a pair of sequences, their output is highly relevant to researchers and can be used to study functional, structural, or evolutionary relationships between sequences [110]. On the other hand, alignment-free methods, broadly defined as any which do not compute base-to-base correspondences, are often capable of only producing a single similarity score between sequences as output. But their tradeoffs are very attractive for many applications: they are much more efficient than alignment in terms of running time and memory requirements – this is especially relevant when dealing with large datasets, like those generated by next-generation sequencing (NGS) applications; and alignment assumes the sequences being compared contain stretches of well-conserved and common regions, which is an assumption often violated in reality, for example when studying viral genomes with a high rate of mutation or comparing sequences from entirely different parts of the tree of life.

2.3 Alignment-free methods

In recent years, a great breadth of alignment-free sequence comparison methods have been proposed. Among these are the conditional Lempel-Ziv complexity as studied in [4, 9, 10, 73, 100, 103, 119, 166], the closely related and more general conditional Kolmogorov complexity [45, 93], the 'measure representation' as proposed in [178, 179], comparisons between Markov models [25, 127], average lengths of maximum common substrings [97, 162], estimates of substitutions or mismatches per site as pioneered by Haubold *et al.* [39, 61, 62, 63], the 'base-base correlation' [104, 105], and distances based on Hasse matrices [161], spectral distortion [126], primitive discrimina-

tion substrings [44], the Burrows-Wheeler similarity [174], normalized central moments [36], nearest-neighbor interactions [181], subword composition [28], prefix codes [38], information correlation [51], the context-object model [176], spaced word frequencies [70, 98], and many more.

Some of these and others have been applied to sequence classification specifically, including methods based on nucleotide correlations [106] or sequence composition (*e.g.* COMET [151] and [177]); on restriction enzyme site distributions, applied to the subtyping of human papillomavirus (HPV), hepatitis B virus (HBV) and HIV-1 (CASTOR [137]); based on the 'natural vector which contains information on the number and distribution of nucleotides in the sequence, applied to the classification of single-segmented [177] and multi-segmented [72] whole viral genomes, as well as viral proteomes [101]; based on neural networks using digital signal processing techniques to yield 'genomic cepstral coefficient' features, applied to distinguishing four different pathogenic viruses [3]; based on different genomic materials (namely DNA sequences, protein sequences, and functional domains) with information based on protein clustering and functional domains, applied to the classification of some viral species at the order, family, and genus levels [165].

2.4 k-mer-based methods

One particular class of methods for alignment-free sequence comparison well-represented in the literature are those based on the frequencies of substrings of length k (known as k-mers or k-words).

Blaisdell was the first to use such methods, reporting success in constructing accurate phylogenetic trees for mammal alpha and beta-globin genes [15] and several other coding and non-coding DNA sequences [14]. Many au-

thors [11, 12, 22, 37, 52, 84, 85, 88, 89] have studied k-mer bias patterns and found that the excess and scarcity of specific k-mers, across a variety of different DNA sequence types (including viral DNA in [22]), can be explained by factors such as physical DNA/RNA structure, mutational events, and some prokaryotic and eukaryotic repair and correction systems. In addition, Karlin *et al.* found that k-mer proportions can play the role of a *genomic signature* – that is, a specific quantitative characteristic of a sequence that is pervasive along the genome of the same organism, while being dissimilar for sequences originating from different organisms. These two findings give intuition and justification as to why the information in k-mer occurrence patterns is suitable as a classifiable feature.

Typically, k-mer frequency or proportion vectors are paired together with a distance function in order to give a method capable of measuring the quantitative similarity of any pair of sequences. Common choices of distance include the Manhattan distance originally proposed by Burge *et al.* [22], as seen e.g. in [16, 17, 23, 53, 77, 83, 87, 88, 121, 132, 153]; the weighted or standardized Euclidean distance, as seen e.g. in [26, 33, 65, 92, 123, 145, 171, 172, 175]; and the Jensen-Shannon distance proposed by Sims *et al.* [148], as seen e.g. in [147, 148, 164, 170]. These distances and others have been compared and benchmarked in [31, 32, 57, 60, 68, 69, 75, 81, 173], and detailed reviews of the literature can be found in [18, 86, 116, 144, 149, 163, 182].

k-mer frequency or proportion vectors have been used to perform supervised classification, albeit often with relatively small datasets. For instance, these vectors have been used to subtype influenza and classify polyoma and rhinovirus fragments [46], to predict HPV genotypes [154, 155], to classify whole bacterial genomes to their corresponding taxonomic groups at different levels [167], to classify whole eukaryotic mitochondrial genomes [112, 113, 114,

115], to classify 27 microbial nuclear DNA sequences [132], to automatically learn a distance function for classifying a set of 1076 microbial genomes [123], to classify hundreds of thousands of short (less than 10,000 base pairs long) prokaryote sequences into different phylogenetic groups [1, 2, 108], to distinguish very short samples of the *E.coli* and yeast genomes [124], to classify short bacterial genome fragments from 28 species [142], to classify longer bacterial genome fragments from 118 species [158], to classify some archaeal and bacterial classes [41], and to classify short splicing-related sequence fragments [117, 128].

2.5 Our approach

KAMERIS is a supervised classification method based on k-mer proportion vectors. In this study and as opposed to other studies described, we do not use just one or a small number of datasets but dozens of datasets covering a large breadth and depth of genomic sequence data. This allows us to give more evidence and be more confident of our performance and accuracy on a wide range of highly biologically-relevant classification tasks, which is not often done with other algorithms and methods.

KAMERIS is fully open-source with all code available on GitHub at the following URL: `https://github.com/stephensolis/kameris`, is standalone and easy to run on any local computer, and is available to all users under a permissive open-source license. This is as opposed to other tools, some of which are closed-source and available only as a web interface or are sold under a commercial license for non-academic users. The fact that KAMERIS is standalone makes it possible for researchers to save and reproduce results with a precise version of the software, and to avoid sending potentially sensitive

data to a remote web server. Further, our open-source implementation makes it easy for researchers to see all technical details of our method, making our tool highly transparent. Those benefits do not come at the expense of classification accuracy, however, since we perform head-to-head evaluations on the tasks of HIV-1 virus subtyping and bacterial taxonomic classification with a number of competing state-of-the-art software solutions, and show that we match or exceed all other tested software in terms of accuracy and speed.

Kameris is very flexible and every experiment presented in this work can be reproduced by following the step-by-step instructions at https://github.com/stephensolis/kameris-experiments. Kameris permits the user to specify their choice of classification algorithm, value of k, cross-validation and dimentionality reduction parameters as described in Chapter 3 when training a model.

Finally, as will be seen in the following chapter, our method is remarkably simple, relying only on the counting of k-mers and well-known supervised classification algorithms. This is as opposed to other much more complex methods, some of which use complex distance functions or correlation metrics and often deep domain-specific biological knowledge – our method is simpler and more computationally efficient than some others without loss of accuracy.

Chapter 3

An open-source k-mer based machine learning tool for fast and accurate subtyping of HIV-1 genomes[1]

3.1 Introduction

Subtype classification is an important and challenging problem in the field of virology. Subtypes (also termed clades or genotypes) are a fundamental unit of virus nomenclature (taxonomy) within a defined species, where each subtype corresponds to a cluster of genetic similarity among isolates from the global population. Defined subtype references for hepatitis C virus, for example, can diverge by as much as 30% of the nucleotide genome sequence [146], but there is no consistent threshold among virus species. Many virus subtypes

[1] A version of this chapter was accepted for publication in PLoS One (S. Solis-Reyes, M. Avino, A. Poon, and L. Kari)

are clinically significant because of their associations with variation in pathogenesis, rates of disease progression, and susceptibility to drug treatments and vaccines [156]. For example, the HIV-1 subtypes originated early in the history of the global pandemic [169] and have diverged by about 15% of the nucleotide genome sequence [78]. Rates of disease progression vary significantly among HIV-1 subtypes and classifying newly diagnosed infections by their genetic similarity to curated reference subtypes [139] is a recommended component for the clinical management of HIV-1 infection [27, 67]. Consequently, a number of algorithms have been developed for the automated determination of HIV-1 subtypes from genetic sequence data [129, 130, 151].

Today, there are important practical considerations that HIV-1 subtyping algorithms should meet. These include:

1. **High Accuracy and Performance:** The cost of sequencing is rapidly decreasing and the amount of sequence data increasing due to next-generation sequencing (NGS) technologies. Thus, in addition to being accurate, software must be computationally fast and scalable in order to handle rapidly growing datasets.

2. **Data Security and Privacy:** Policy, legal, and regulatory issues can prohibit patient sequence data from being transmitted to an external server on the Internet. In addition, concerns around privacy policies and the possibility of data breaches can cause issues for researchers and clinicians. For these reasons, software should be made available in an offline, standalone version.

3. **Transparency:** With closed-source or proprietary software, it can be impossible to determine precisely how classification determinations are made. An open-source implementation gives full visibility into all aspects

of the classification process.

4. **Reproducibility:** Relying on an externally-hosted service can make it impossible to determine which version of the software has been used to generate subtype classifications. This makes it difficult to guarantee that classification results can be reproduced, and reproducibility is generally recognized as a necessary component of clinical practice.

In our effort to develop a general sequence classification method satisfying the above considerations, we propose a simple, intuitive, general-purpose, highly-efficient technique based on k-mer proportion vectors for supervised nucleotide sequence classification, and we release an open-source software implementation of this method (designated KAMERIS) under a permissive open-source license.

3.1.1 Alignment-free subtyping

Most subtype classification methods for HIV-1 require the alignment of the input sequence against a set of predefined subtype reference sequences [95], which enables the algorithm to compare homologous sequence features [40, 50, 143]. For example, the NCBI genotyping tool [140] computes BLAST similarity scores against the reference set for sliding windows along the query sequence. Other methods such as REGA [129] and SCUEAL [130] reconstruct maximum likelihood phylogenies from the aligned sequences: REGA (version 3.0) reconstructs trees from sliding windows of 400 bp from the sequence alignment and quantifies the confidence in placement of the query sequence within subtypes by bootstrap sampling (bootscanning) [141]. Alignment-based methods are relatively computationally expensive, especially for long sequences; the heuristic methods require a number of *ad hoc* settings, such as the penalty for opening a gap; and alignment method may not perform

well on highly-divergent regions of the genome. To address these limitations, various alignment-free classification methods have been proposed. Some of them make use of nucleotide correlations [106], or sequence composition (*e.g.* COMET [151] and [177]). Other methods include those based on restriction enzyme site distributions, applied to the subtyping of human papillomavirus (HPV), hepatitis B virus (HBV) and HIV-1 (CASTOR [137]); based on the "natural vector" which contains information on the number and distribution of nucleotides in the sequence, applied to the classification of single-segmented [177] and multi-segmented [72] whole viral genomes, as well as viral proteomes [101]; based on neural networks using digital signal processing techniques to yield "genomic cepstral coefficient" features, applied to distinguishing four different pathogenic viruses [3]; and based on different genomic materials (namely DNA sequences, protein sequences, and functional domains), applied to the classification of some viral species at the order, family, and genus levels [165].

3.1.2 k-mer-based classifiers

The use of k-mer (substrings of length k) frequencies for phylogenetic applications started with Blaisdell, who reported success in constructing accurate phylogenetic trees from several coding and non-coding nuclear genomes sequences [14] and some mammalian alpha and beta-globin genes [15]. Other authors [22, 52, 84, 88, 89] have observed that the excess and scarcity of specific k-mers, across a variety of different DNA sequence types (including viral DNA in [22]), can be explained by factors such as physical DNA/RNA structure, mutational events, and some prokaryotic and eukaryotic repair and correction systems. Typically, k-mer frequency or proportion vectors are paired together with a distance function in order to measure the quantitative similarity be-

tween any pair of sequences. Studies measuring quantitative similarity between DNA sequences from different sources have been performed, for instance using the Manhattan distance [23, 87], the weighted or standardized Euclidean distance [145, 172], and the Jensen-Shannon distance [147, 148]. Applications of these distances and others have been compared and benchmarked in [32, 60, 81, 173], and detailed reviews of the literature can be found in [18, 116, 163, 182].

In the context of viral phylogenetics, k-mer frequency or proportion vectors paired with a distance metric have been used to construct pairwise distance matrices and derive phylogenetic trees, *e.g.*, dsDNA eukaryotic viruses [170], or fragments from Flaviviridae genomes [92]. Other studies have investigated the multifractal properties of k-mer patterns in HIV-1 genomes [120], and the changes in dinucleotide frequencies in the HIV genome across different years [121]. We used k-mer proportion vectors to train supervised classification algorithms. Similar approaches have previously been explored (with different classifiers than those used here), for example to subtype Influenza and classify Polyoma and Rhinovirus fragments [46], to predict HPV genotypes [154, 155], to classify whole bacterial genomes to their corresponding taxonomic groups at different levels [167], and to classify whole eukaryotic mitochondrial genomes [112, 113, 114, 115].

To evaluate our method, we curated manually-validated testing sets of 'real-world' HIV-1 data sets. We assessed fifteen classification algorithms and conclude that for these data the SVM-based classifiers, multilayer perceptron, and logistic regression achieved the highest accuracy, with the SVM-based classifiers also achieving the lowest running time out of those. We measured classification accuracy and running time for k-mers of length $k = 1 \ldots 10$

and found that $k = 6$ provides the optimal balance of accuracy and speed. Overall, our open-source method obtains a classification accuracy average of 97%, with individual accuracies equal to or exceeding other subtyping methods for most datasets, and processes over 1,500 sequences per second. Our method is also applicable to other virus datasets without modification: we demonstrate classification accuracies of over 90% in all cases for full-length genome data sets of dengue, hepatitis B, hepatitis C, and influenza A viruses.

3.2 Methods

3.2.1 Supervised classification

First, we needed to determine which supervised classification method would be the most effective for classifying virus sequences, using their respective k-mer proportions as feature vectors (numerical representations). We trained each of 15 classifiers (Table 3.2) on a set $S = \{s_1, s_2, \ldots s_n\}$ of nucleotide sequences partitioned into groups $g_1, g_2, \ldots, g_p$. Given as input any new, previously unseen, sequence (*i.e.*, not in the dataset S), the method outputs a prediction of the group g_i that the sequence belongs to, having 'learned' from the training set S the correspondence between the k-mer proportions of training sequences and their groups. The feature vector $F_k(s)$ for an input sequence s was constructed from the number of occurrences of all 4^k possible k-mers (given the nucleotide alphabet $\{A, C, G, T\}$), divided by the total length of s. Any ambiguous nucleotide codes (*e.g.*, *'N'* for completely ambiguous nucleotides) were removed from s before computing $F_k(s)$. As a concrete example, suppose $s = ACTCAGGCA$ and $k = 2$. Then, if we use the arbitrary order $[AA, AC, AG, AT, CA, CC, CG, CT, GA, GC, GG, GT, TA,$

$TC, TG, TT]$ for 2-mers, the k-mer frequency vector for s is $[0, 1, 1, 0, 2, 0, 0, 1, 0, 1, 1, 0, 0, 1, 0, 0]$ and thus $F_k(s) = [0, 0.125, 0.125, 0, 0.25, 0, 0, 0.125, 0, 0.125, 0.125, 0, 0, 0.125, 0, 0]$.

Next, we processed the feature vectors $F_k(s)$ for more efficient use by classifiers. We rescaled the vectors to have a variance of 1, which satisfies some statistical assumptions invoked by several classification methods. In addition, we performed dimensionality reduction using truncated singular value decomposition [54] to reduce the vectors to 10% of the average number of non-zero entries of the feature vectors. This greatly reduces running time for most classifiers while having a negligible effect on classification accuracy.

Finally, we trained a supervised classifier on the vectors $F_k(s)$. Supervised classifiers, in general, can be intuitively thought of as constructing a mapping from the input feature space to another space which in some sense effectively separates each training class. As a concrete example, the support vector machine (SVM) classifier maps the input space to another space of equal or higher dimensionality using a kernel function, and then selects hyperplanes that represent the largest separation between every pair of classes. Those hyperplanes induce a partition on the transformed space which is then used for the classification of new items. We tested fifteen different specific classifier algorithms: 10-nearest-neighbors [5] with Euclidean metric (`10-nearest-neighbors`); nearest centroid, to class mean (`nearest-centroid-mean`) and to class median (`nearest-centroid-median`) [160]; logistic regression with L2 regularization and one-vs-rest as the multiclass generalization (`logistic-regression`) [13]; SVM with the linear (`linear-svm`), quadratic (`quadratic-svm`), and cubic (`cubic-svm`) kernel functions [30]; SVM with stochastic gradient descent learning and linear kernel function (`sgd`) [180]; decision tree with Gini impurity metric (`decision-tree`) [21]; random forest using

decision trees with Gini impurity metric as sub-estimators (`random-forest`) [20]; AdaBoost with decision trees as the weak learners and the SAMME.R real boosting algorithm (`adaboost`) [48, 59]; Gaussian naïve Bayes (`gaussian-naive-bayes`) [24]; linear (`lda`) and quadratic (`qda`) discriminant analysis [49]; and multi-layer perceptron with a 100-neuron hidden layer, rectified linear unit (ReLU) activation function, and the Adam stochastic gradient-based weight optimizer (`multilayer-perceptron`) [66, 91]. We used the implementations of these classifiers in the Python library `scikit-learn` [125] with the default settings.

For some of the results that follow, we required a method for measuring classification accuracy without the need for a separate testing dataset. To do so, we used 10-fold cross-validation, a technique widely used for assessing the performance of supervised classifiers [136]. N-fold cross-validation is performed by taking the given dataset and randomly partitioning it into N groups of equal size. Taking each group in turn, we trained a classifier on the sequences outside of the selected group, and then computed its accuracy from predicting the classes of the sequences in the selected group. The outcome of the cross-validation are N accuracy values for the N distinct, independent training and testing runs. We report the arithmetic mean of those accuracies as the final accuracy measure.

3.2.2 Unsupervised visualization

Supervised classification requires, by definition, a training set consisting of examples of classes determined *a priori*. However, one may wish to explore a dataset where the groups are not necessarily all known. For the problem of virus subtyping for example, one may suspect the existence of a novel subtype or recombinant. To this end, unsupervised data exploration techniques are useful, and herein we also explore the use of Molecular Distance Maps (MoDMaps), previously described in [80, 81, 82], for this purpose. After computing the vectors $F_k(s)$, this method proceeds by first constructing a pairwise distance matrix. In this paper, we use the well-known Manhattan distance [94], defined between two vectors $A = (a_1, \ldots a_n)$ and $B = (b_1, \ldots b_n)$ as being:

$$d_M(A, B) = \sum_{i=1}^{n} |a_i - b_i| .$$

Next, the distance matrix is visualized by classical MultiDimensional Scaling (MDS) [19]. MDS takes as input a pairwise distance matrix and produces as output a 2D or 3D plot, called a MoDMap [79], wherein each point represents a different sequence, and the distances between points approximate the distances from the input distance matrix. As MoDMaps are constrained to two or three dimensions, it is in general not possible for the distances in the 2D or 3D plot to match exactly the distances in the distance matrix, but MDS attempts to make the difference as small as possible.

3.2.3 Implementation

We have developed a software package called KAMERIS which implements our method. It can be obtained from https://github.com/stephensolis/

kameris, and may be used on Windows, macOS, and Linux. KAMERIS is implemented in Python, with the feature vector computation parts implemented in C++ for performance. It is packaged so as to have no external dependencies, and thus is easy to run. The package has three different modes: first, it can train one or more classifiers on a dataset and evaluate cross-validation performance; second, it can summarize training jobs, computing summary statistics and generating MDS plots; and third, it can classify new sequences on already-trained models. More information, including usage and setup instructions, can be found at https://github.com/stephensolis/kameris. All running time benchmarks of our software were performed on an Amazon Web Services (AWS) r4.8xlarge instance with 16 physical cores (32 threads) of a 2.3GHz Intel Xeon E5-2686 v4 processor. We also note that many of the implementations of the classifier algorithms we use are single-threaded and that performance can almost certainly be substantially improved by using parallelized implementations.

3.2.4 Datasets

In this paper, a variety of different datasets were used to validate the performance of the method. Straightforward reproducibility of results was a priority in the design of this study, and to that end, every sequence and its metadata from every dataset referenced here can be retrieved from our GitHub repository at https://github.com/stephensolis/kameris-experiments. Further, instructions for using KAMERIS to replicate the experiment results are available in Appendix A.

In some cases, these datasets had few examples for some classes. Training on classes with very few examples would unfairly lower accuracy since the

classifier does not have enough information to learn, so we wish to omit such classes from our analysis. However, the minimum number of examples per class to achieve proper training of a classifier is difficult to estimate; this number is known to be dependent on both the complexity of the feature vectors and characteristics of the classifier algorithm being used [74, 135]. Since we vary both k and the classifier algorithms in this study, this makes it especially challenging to empirically determine an adequate minimum class size. Here, we arbitrarily selected 18 as our minimum, so we omitted from analysis any subtype with fewer than 18 sequences. It may be that specific values of k and some classifier algorithms work well in scenarios with very small datasets, and we leave this as an open question.

Primary dataset

The primary dataset used was the full set of HIV-1 genomes available from the Los Alamos (LANL) sequence database, accessible at https://www.hiv.lanl.gov/components/sequence/HIV/search/search.html. In this database, the option exists of using full or partial sequences – in our analysis, we consider both full genomes and just the coding sequences of the *pol* gene. For the set of whole genomes, the query parameters "virus: HIV-1, genomic region: complete genome, excluding problematic" were used; this gave a total of 6625 sequences with an average length of 8970 bp. For the set of *pol* genes, the query parameters "virus: HIV-1, genomic region: Pol CDS, excluding problematic" were used; this gave a total of 9270 sequences with an average length of 3006 bp. In both cases, the query was performed on May 18, 2017, and at the time, the LANL database reported a last update on May 6, 2017. After removing small classes (see preceding section), this dataset contained 25 subtypes and circulating recombinant forms (CRFs) for the set

of whole genomes, and 26 for the set of *pol* genes. The list of subtypes for this dataset and all other datasets described here are available in Appendix B. This dataset was used to determine the best value of k, the best classifier algorithm, to compare the performance of whole genomes with *pol* gene sequences only, and to produce the MoDMaps of HIV-1. In those experiments, cross-validation was used to randomly draw training and testing sets from the dataset.

Evaluation datasets

To evaluate classifiers trained on HIV-1 sequences and subtype annotations curated by the LANL database, we needed testing sets but wanted to avoid selecting them from the same database. We manually searched the GenBank database for large datasets comprising HIV-1 *pol* sequences collected from a region with known history of a predominant subtype, and evaluated the associated publications to verify the characteristics of the study population (Table 3.1). After selection of the datasets, we wished to obtain labels without relying on another subtyping method. To do so, first we made use of the known geographic distribution of HIV-1 subtypes, where specific regions are predominantly affected by one or two particular subtypes or circulating recombinant forms due to historical 'founding' events [157]. Next, we screened each dataset using a manual phylogenetic subtyping process to verify subtype assignments against the standard reference sequences. This was done, essentially, by reconstructing phylogenetic trees to identify possible subtype clusters. A cluster was identified as a certain subtype if it included a specific subtype reference sequence we had initially provided in our datasets. Thus, the first step was to download the most recent set of subtypes reference sequences for the HIV-1 *pol* gene at the LANL database, accessible at `https:`

//www.hiv.lanl.gov/content/sequence/NEWALIGN/align.html [99].

We loaded the resulting FASTA file in the eleven datasets from Table 3.1. We then aligned the datasets with MUSCLE v3.8.425 [42], implemented in AliView 1.19-beta-3 [96], where we also visually inspected the alignments. To avoid overfitting, we searched for the nucleotide model of substitution that was best supported by each dataset using the Akaike Information Criterion (AIC) in jModeltest v2.1.10 [34]. For the dataset US.Wolf2017, the large number of sequences precluded this model selection process, so we chose a General Time Reversible model incorporating an invariant sites category and a gamma distribution to model rate variation among the remaining sites (GTR+I+G); this parameter-rich model is often supported by large HIV-1 data sets, and was similar to the model selected by the authors in the original study [168]. Phylogenetic trees were reconstructed by maximum likelihood using PHYML v20160207 [56] with a related bootstrap support analysis. The resulting trees were visualized and their relative sequences were manually annotated in FigTree v1.4.3 [134].

Table 3.1: **Statistics for the manually curated testing datasets.** The first author, year, and reference number for the publication associated with each data set is listed under the 'Source' column heading. The historically most prevalent HIV-1 subtype(s) is indicated under the 'Subtype' column heading.

Source	**Country**	**Subtype**	**Count**	**Sequence length (nt)**		
				Average	Min.	Max.
Nadai (2009) [111]	Haiti	B	66	1024.0	1024	1025
Niculescu (2015) [118]	Romania	F	97	1301.2	1257	1302
Paraschiv (2017) [122]	Romania	F	86	1295.9	1164	1299
Rhee (2017) [138]	Thailand	CRF01_AE	282	703.8	633	756
Sukasem (2007) [152]	Thailand	CRF01_AE	221	286.4	270	288
Eshleman (2001) [43]	Uganda	A/D	102	1261.2	1260	1302
Ssemwanga (2012) [150]	Uganda	A/D	72	1025.0	1025	1025
Wolf (2017) [168]	USA	B	1653	1020.8	868	1080
TenoRes (2016) [55]	South Africa	C	102	1001.4	921	1209
van Zyl (2017) [183]	South Africa	C	59	1056.7	1002	1070
Huang (2003) [71]	N/A	N/A	44	1189.9	1187	1190
Overall			2784	960.4	270	1302

In order to benchmark performance on this manually curated testing dataset, we required a separate training dataset. Since the subtype annotations from the full set of HIV-1 genomes in the LANL database are typically given by individual authors using unknown methods, they may be incorrect at times, potentially negatively impacting classification performance. Thus, we trained our classifier on the subset of HIV-1 *pol* sequences from the 2010 Web alignment from the LANL database, accessible at https://www.hiv.lanl.gov/content/sequence/NEWALIGN/align.html. This Web alignment dataset

is a more curated set of *pol* sequences, and is more likely to be correctly annotated. Specifically, we selected 'Web' as the alignment type, 'HIV1/SIVcpz' as organism, 'POL' as 'Pre-defined region of the genome' under 'Region', 'All' as subtype, 'DNA', and '2010' as the year. Any Simian Immunodeficiency Virus (SIV) sequences were manually removed from the query results. This gave a total of 1979 sequences, containing 15 subtypes or CRFs after removal of small classes.

Other datasets

For another experiment, we generated a set of synthetic HIV-1 sequences by simulating the molecular evolution of sequences derived from the curated HIV-1 subtype references. To do so, we used a modified version of the program INDELible [47], assigning one of the subtype reference sequences to the root of a 'star' phylogeny with unit branch lengths and 100 tips. The codon substitution model parameters, including the transition-transversion bias parameter and the two-parameter gamma distribution for rate variation among sites, were calibrated by fitting the same type of model to actual HIV-1 sequence data [131]. We adjusted the 'treelength' simulation parameter to control the average divergence between sequences at the tips.

Finally, we performed experiments with dengue, influenza A, hepatitis B, and hepatitis C virus sequences. The dengue and influenza sequences were retrieved from the National Center for Biotechnology Information (NCBI) Virus Variation sequence database on August 10, 2017. The dengue virus sequences were accessed from https://www.ncbi.nlm.nih.gov/genomes/VirusVariation/Database/nph-select.cgi?taxid=12637 with the query options "Nucleotide", "Full-length sequences only", and "Collapse identical sequences" for a total of 4893 sequences with an average length of 10585 bp. Influenza se-

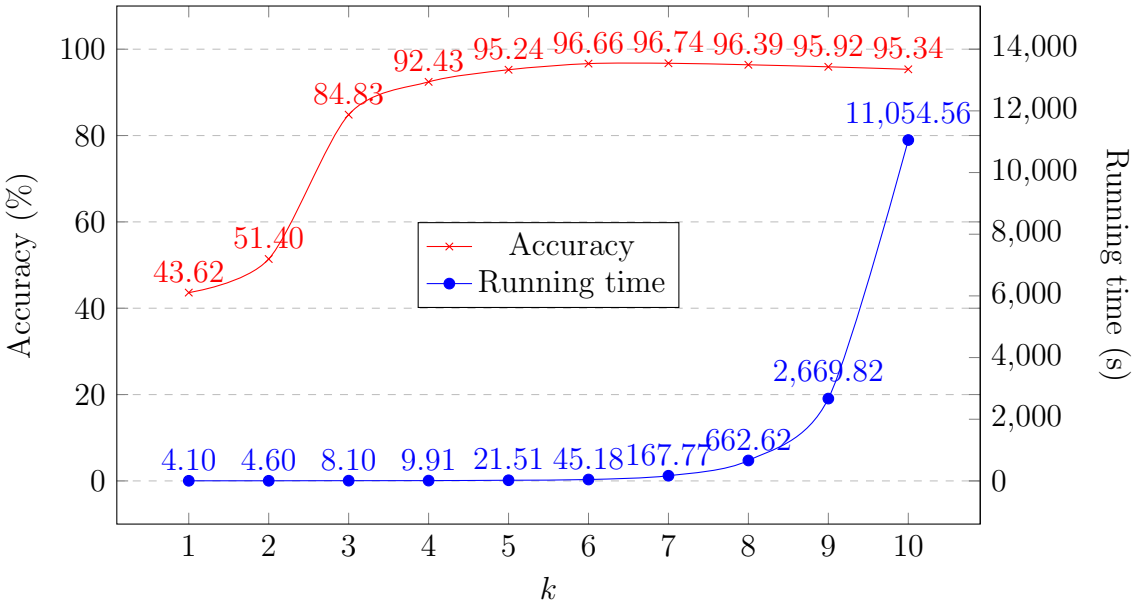
quences were accessed from https://www.ncbi.nlm.nih.gov/genomes/FLU/Database/nph-select.cgi?go=genomeset with the query options "Genome sets: Complete only", and "Type: A" for a total of 38215 sequences with an average length of 13455 bp. Hepatitis B sequences were retrieved from the Hepatitis B Virus Database operated by the Institut de Biologie et Chimie des Proteines (IBCP), accessible at https://hbvdb.ibcp.fr/HBVdb/HBVdbDataset?seqtype=0, on August 10, 2017 for a total of 5841 sequences with an average length of 3201 bp. Finally, hepatitis C sequences were retrieved from the Los Alamos (LANL) sequence database, accessible at https://hcv.lanl.gov/components/sequence/HCV/search/searchi.html, on August 10, 2017, using the query options "Excluding recombinants", "Excluding 'no genotype'", "Genomic region: complete genome", and "Excluding problematic" for a total of 1923 sequences with an average length of 9140 bp. After removal of small classes, our data comprised 4 subtypes of dengue virus, 12 subtypes of hepatitis B, 6 subtypes of hepatitis C, and 56 subtypes of influenza type A.

3.3 Results

Our subtype classification method has two main parameters that may be varied: namely, the specific classifier to be used, and the value k of the length of the k-mers to count when producing feature vectors. We begin with the full set of full-length HIV-1 genomes from the LANL database, and we perform a separate 10-fold cross-validation experiment for each of the fifteen classifiers listed in the Methods section, and all values of k from 1 to 10, that is, 160 independent experiments in total. For each value of k, we plot the highest accuracy obtained by any classifier as well as the average running time over the classifiers, see Figure 3.1. We observe that $k = 6$ achieves a

good balance between classifier performance and accuracy, so at $k = 6$, we list the accuracy obtained by each classifier and its corresponding running time, see Table 3.2. As can be seen, the SVM-based classifiers, multilayer perceptron, and logistic regression achieve the highest accuracy, with the SVM-based classifiers achieving also the lowest running time out of those.

Figure 3.1: **Highest accuracy score and average running time across all fifteen classifiers, at different values of k, for the full set of 6625 whole HIV-1 genomes from the LANL database.**



Since it is typical to have only partial genome sequences available, we repeat the same 10-fold cross-validation at $k = 6$, with the linear SVM classifier, this time with the set of all *pol* genes from the LANL database. We find that the accuracy changes from 96.49% (full-length genomes) to 95.68% (*pol* gene sequences), indicating that the use of partial genomes does not substantially reduce classification performance. Further, we expect that the inclusion of recombinant forms should lower accuracy, since it requires the classifier to accurately distinguish them from their constituent 'pure' subtypes. To test this, we repeat the same 10-fold cross-validation at $k = 6$ and with the linear

Table 3.2: **Accuracy scores and running times for each of the fifteen classifiers at $k = 6$, for the full set of 6625 whole HIV-1 genomes from the LANL database.**

Classifier	Accuracy	Running time
`cubic-svm`	96.66%	44.44s
`quadratic-svm`	96.59%	44.52s
`linear-svm`	96.49%	44.23s
`multilayer-perceptron`	95.49%	53.92s
`logistic-regression`	95.32%	88.18s
`10-nearest-neighbors`	93.97%	31.92s
`nearest-centroid-median`	93.95%	22.21s
`nearest-centroid-mean`	93.84%	21.90s
`decision-tree`	93.53%	49.99s
`random-forest`	93.07%	31.35s
`sgd`	91.10%	24.24s
`gaussian-naive-bayes`	87.75%	22.39s
`lda`	77.76%	24.46s
`qda`	75.13%	26.57s
`adaboost`	64.85%	147.24s

SVM classifier, with the set of all full-length genomes from the LANL database, this time omitting the 17 classes of recombinant forms and leaving only the 9 classes of pure subtypes. We find that the accuracy increases from 96.49% (including recombinants) to 99.64% (omitting recombiants), and in fact only 3 sequences are misclassified in the latter case.

The sequences present in the LANL database are curated to be representative of global HIV-1 diversity, and therefore high classification accuracies on that dataset are, to some extent, to be expected. In order to perform a more challenging benchmark on our algorithm, we compute its accuracy on the eleven selected testing datasets of *pol* gene fragments from Table 3.1, after training with the set of whole *pol* genes from the LANL 2010 web alignment. Based on the previous performance measurements, we use the linear SVM classifier and $k = 6$. We also perform the same accuracy measurement with four other state-of-the-art HIV subtyping tools: CASTOR, COMET,

SCUEAL, and REGA, and show the results in Table 3.3. In sum, our method (KAMERIS) comes within a few percent of the best tools in all cases, and has the highest average accuracy (both unweighted, and weighted by the number of sequences in each set).

Running time is another important performance indicator, so we also compare the performance of these five tools for the dataset of van Zyl et al. [183], and the four fastest tools for all datasets together (see Table 3.4). We observe that our tool matches or outperforms the competing state-of-the-art. Note that, for these comparison experiments, CASTOR, COMET, SCUEAL, and REGA were run from their web-based interfaces, and therefore the exact specifications of the machines running each program could not be determined. For this reason, the running times presented here should be taken as rough order-of-magnitude estimates only.

Overall, these experiments demonstrate our method is nearly identical in both accuracy and running time to the top third-party tool, COMET. Our tool differs from COMET in that it is open-source and freely available for commercial use, and is available in a standalone application which can be run on any computer, while COMET is closed-source and freely available for non-commercial research use only, and is publicly available only in a web-based system.

Table 3.3: **Classification accuracies for all tested HIV-1 subtyping tools, for each testing dataset from Table 3.1; average accuracy both with and without weighting datasets by the number of sequences they contain.**

Source	Kameris	COMET	CASTOR	SCUEAL	REGA
Nadai (2009) [111]	100.0%	100.0%	81.8%	92.4%	86.4%
Niculescu (2015) [118]	95.9%	96.9%	75.3%	94.8%	100.0%
Paraschiv (2017) [122]	91.9%	73.3%[1]	46.5%	68.6%	87.2%
Rhee (2017) [138]	94.0%	95.4%	0.4%	75.9%	12.8%[1]
Sukasem (2007) [152]	90.0%	91.0%	0.9%	64.3%	8.1%[2]
Eshleman (2001) [43]	88.5%	90.6%	4.2%	84.4%	90.6%
Ssemwanga (2012) [150]	88.3%	90.0%	0.0%	73.3%	95.0%
Wolf (2017) [168]	99.8%	99.8%	61.1%	99.3%	98.2%
TenoRes (2016) [55]	99.0%	99.0%	28.4%	99.0%	100.0%
van Zyl (2017) [183]	94.9%	93.2%	57.6%	93.2%	94.9%
Huang (2003) [71]	95.2%	97.6%	19.0%	81.0%	95.2%
Average (unweighted)	94.3%	93.3%	34.1%	84.2%	78.9%[2]
Average (weighted)	97.1%	96.9%	45.1%	91.2%	81.4%[2]

[1] In this case, a substantial number of sequences that were classified as subtype A by REGA and our method were labeled unclassified subtypes (U) by COMET. In an HIV-1 phylogeny, subtype U sequences tend to be assigned a basal position (near the root) within the subtype A clade, suggesting that these sequences may be unrecognized variants or complex recombinants of subtype A.

[2] These low accuracies are primarily caused by REGA misclassifying many CRF01 sequences as subtype A, and subtype A is mostly equivalent to CRF01 in the *pol* region. If CRF01 and A were treated as equivalent, these accuracies would be 97.9% and 86.4% for the Rhee and Sukasem datasets, respectively, and unweighted and weighted averages of 93.8% and 96.2%, respectively.

Table 3.4: **Approximate running times for all tested subtyping tools, for the dataset of van Zyl et al. [183] and all datasets listed in Table 3.3.** The van Zyl dataset was chosen at random for this purpose.

Tool	Running time for the van Zyl dataset	Running time for datasets from Table 3.3
Kameris	less than 2 seconds	16 seconds
COMET	less than 2 seconds	14 seconds
CASTOR	3 seconds	46 seconds
SCUEAL[1]	18 minutes	8 hours
REGA[1]	31 minutes	19 hours

[1] The REGA and SCUEAL web servers have limits of 1000 and 500 sequences per run, respectively. Thus, 3 batches of sequences were needed for REGA, and 6 batches for SCUEAL to classify all sequences. COMET, CASTOR, and our tool have no such limits.

So far, we have only discussed supervised classification, and we have presented promising results for our approach. However, supervised classification requires data with known labels, which can be problematic considering that the rapid rates of mutation and recombination of viruses (particularly HIV-1) can lead to novel strains and recombinant forms emerging quickly. Unsupervised data exploration tools can help address this problem. To demonstrate, we take the set of all whole genomes from the LANL database and produce a MoDMap, visualizing their interrelationships, based on the Manhattan distance matrix obtained by computing all pairs of k-mer proportion vectors (see Methods section), for 9 different pure subtypes or groups (Figure 3.2), and just subtypes A, B, and C (Figure 3.3). As can be seen, based on these distances, the points in the plots are grouped according to known subtypes, and indeed it can be seen that subtypes A1 and A6 group together, and as well B and D group together, as could be expected.

Figure 3.2: **MoDMap of 4373 full-length HIV-1 genomes of 9 different pure subtypes or groups, at $k = 6$.**

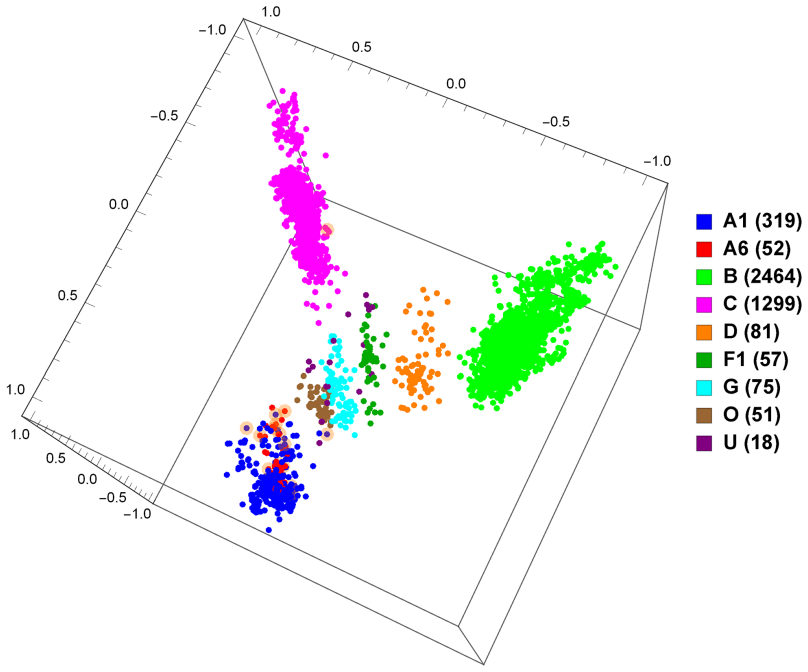
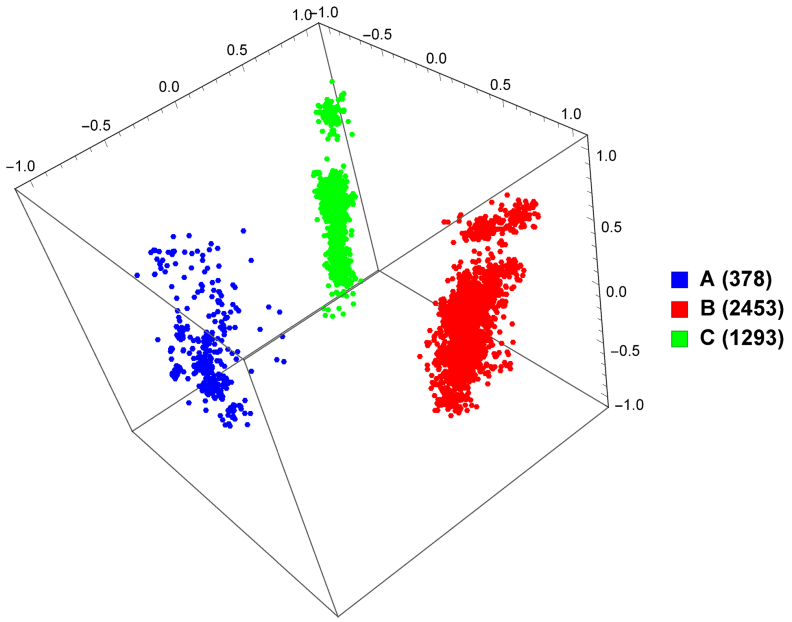


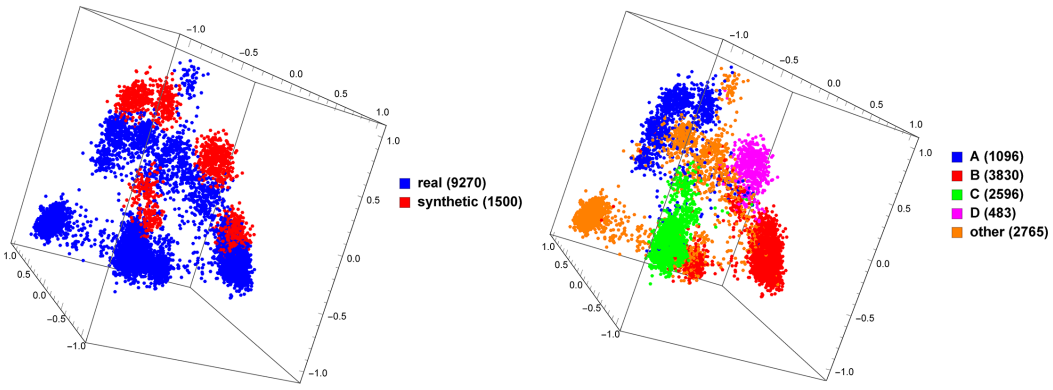
Figure 3.3: **MoDMap of 4124 full-length HIV-1 genomes of subtypes A, B, and C, at $k = 6$.**



Synthetic data has been useful in the study of viral species such as HIV-1, because a ground-truth classification is known for synthetic sequences without ambiguity. However, one may wonder how well such synthetic sequences model natural ones. We attempt to measure this by training a classifier on natural and synthetic HIV-1 sequence data – if natural and synthetic sequences cannot be distinguished, one may conclude that the simulation is realistic. For the 'natural' class we use the set of all *pol* genes from the LANL database, and for the 'synthetic' class we use 1500 synthetic *pol* genes produced as detailed previously, and we perform a 10-fold cross-validation at $k = 6$ and with the linear SVM classifier. We obtain an accuracy of 100%, meaning that the classifier can distinguish natural from synthetic sequences with perfect accuracy. This suggests that synthetic sequence data should be used with caution, since this result indicates it may not be perfectly representative of natural sequence data – specifically, our result suggests there is some characteristic of the syn-

thetic sequences which differs from the natural sequences, which our method is able to recognize and use. We explore this further by generating a MoDMap, as seen in Figure 3.4. Interestingly, even though our supervised classifiers succeeded to discriminate between real and synthetic sequences with an accuracy of 100%, the approach using distances between k-mer proportion vectors results in the natural and synthetic sequences of specific subtypes grouping together, indicating that the synthetic sequences have some features that relate them to the corresponding natural sequences of the same subtype.

Figure 3.4: **MoDMap of 9270 natural HIV-1 *pol* genes vs. 1500 synthetically generated HIV-1 *pol* genes of various subtypes.** The same plot is colored on the left by type (natural and synthetic) and on the right by HIV-1 subtype.



3.4 Discussion

The k-mer based supervised classification method we propose in this paper has several advantages compared to other popular software packages for the classification of virus subtypes. First, we have shown on several manually-curated data sets that k-mer classification can be highly successful for rapid and accurate HIV-1 subtyping relative to the current state-of-the-art. Furthermore, releasing our method as an open-source software project confers

significant advantages with respect to data privacy, transparency and reproducibility. Other subtyping algorithms such as REGA [35] and COMET [151] are usually accessed through a web application, where HIV-1 sequence data is transmitted over the Internet to be processed on a remote server. This arrangement is convenient for end-users because there is no requirement for installing software other than a web browser. However, the act of transmitting HIV-1 sequence data over a network may present a risk to data privacy and patient confidentiality – concerns include web applications neglecting to use encryption protocols such as TLS, or servers becoming compromised by malicious actors. As a concrete example, the webserver hosting the first two major releases of the REGA subtyping algorithm [35] was recently compromised by an unauthorized user (last access attempt on November 27, 2017). In contrast, our implementation is available as a standalone program, without any need to transmit sequence data to an external server, eliminating those issues. In addition, our implementation is released under a permissive open-source license (MIT). In contrast, REGA [129] and COMET [151] are proprietary 'closed-source' software, making it impossible to determine exactly how subtype predictions are being generated from the input sequences.

Relying on a remote web server to process HIV-1 sequence data makes it difficult to determine which version of the software has been used to generate subtype classifications, and by extension difficult to guarantee that classification results can be reproduced. There is growing recognition that tracking the provenance (origin) of bioinformatic analytical outputs is a necessary component of clinical practice. For example, the College of American Pathologists recently amended laboratory guidelines on next-generation sequence (NGS) data processing to require that: "the specific version(s) of the bioinformatics pipeline for clinical testing using NGS data files are traceable for each patient

report" [8]. In contrast to other tools, our standalone package makes it easy to use exactly the desired version of the software and thus enables precise reproducibility.

We now discuss some limitations of our approach. Like many machine learning approaches, our method does not provide an accessible explanation as to why a DNA sequence is classified a certain way, compared to a more traditional alignment-based method. In some sense, the classifiers act more as a black box, without providing a rationale for their results. Another issue is our requirement for a sizable, clean set of training data. As opposed to an alignment-based method that could function with even a single curated reference genome per class, machine learning requires several examples per training class, as discussed previously, to properly train. Finally, one issue common to any HIV-1 subtyping tool is the fact that recombination and rapid sequence divergence can make subtyping difficult, especially in cases where the recombinant form was not known at the time of training. Other tools are capable of giving a result of 'no match' to handle ambiguous cases, but our method always reports results from the classes used for training.

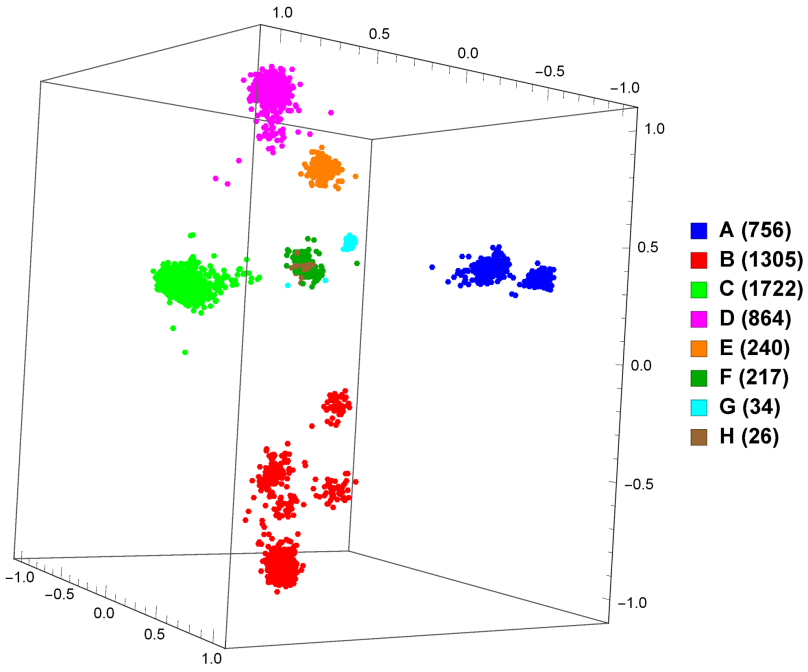
To more clearly demonstrate this last issue, we generate a random sequence of length 10,000 with equal occurrence probabilities for A, C, G, and T, and we ask the five subtyping tools evaluated in our study to predict its HIV-1 subtype. As expected, REGA gives a result of 'unassigned' and SCUEAL reports a failure to align with the reference. Our tool reports subtype 'U' with 100% confidence, CASTOR predicts HIV-1 group 'O' with 100% confidence, and COMET reports SIV_{CPZ} (simian immunodeficiency virus from chimpanzee) with 100% confidence. These outcomes are consistent with the disproportionately large genetic distances that separate HIV-1 group O and SIV_{CPZ} from HIV-1 group M – a line drawn from a random point in sequence

space is more likely to intersect the branch relating either of these distant taxa to group M. Similarly, branches leading to subtype U sequences tend to be longer and to intersect the HIV-1 group M tree at a basal location[2]. This artificial example implies that real HIV-1 sequences that do not readily fit into any of the defined subtypes or circulating recombinant forms may result in incorrect predictions with misleadingly high confidence scores.

In spite of these limitations, our method not only matches or improves upon current HIV-1 subtyping algorithms, but it should also be broadly applicable to any DNA sequence classification problem, including other virus subtyping problems. To demonstrate this, we use the same method (with k set to 6 and a linear SVM classifier) and 10-fold cross-validation to measure the accuracies for classifying dengue, hepatitis B, hepatitis C, and influenza type A virus full-length genomes (described in the Datasets section) to their respective reference subtypes. Overall, we obtain accuracies of 100% for dengue virus, 95.81% for hepatitis B virus, 100% for hepatitis C virus, and 96.68% for influenza A virus. We also provide a MoDMap visualization of the subtypes of hepatitis B, as seen in Figure 3.5. This plot displays not only clear separation between subtypes but also structure within subtypes A and B, which would be an interesting target for future study.

[2]HIV-1 subtype U does not comprise a distinct clade. Rather, the LANL database labels sequences as 'U' when they belong to a lineage not meeting the criteria required for a designation as a subtype [139]. However, practical but anecdotal experience suggests subtype U sequences are typically basal.

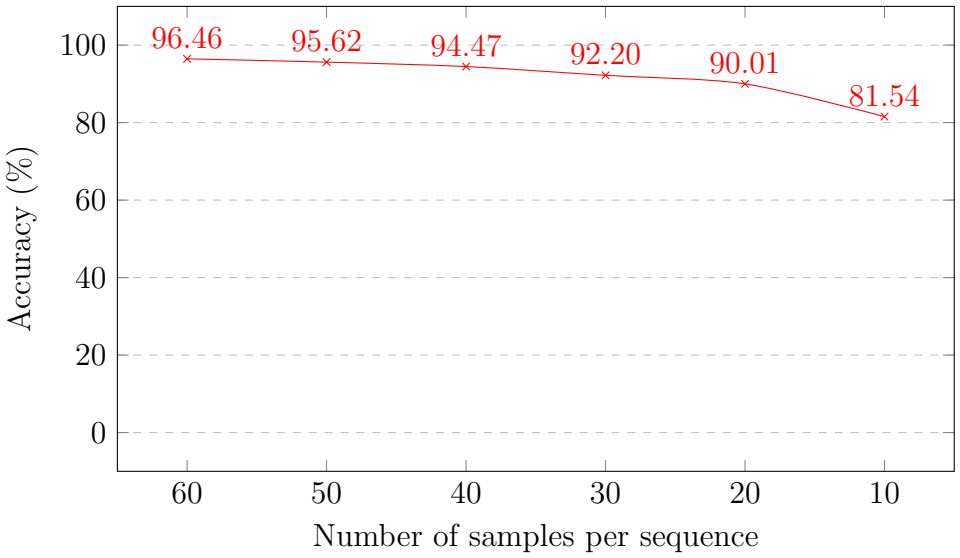
Figure 3.5: **MoDMap of 5164 whole hepatitis B genomes of 6 different pure subtypes.**



In all the experiments presented above, we use whole assembled genomes or gene sequences. However, next-generation sequencing (NGS) technologies produce as output short reads, often of length 150 to 300 base pairs, and computationally-intensive assembly is required to produce contiguous sequences. Usefully, our method works equally well on short reads, without any requirement for assembly. To validate this, we begin with the full set of whole HIV-1 genomes from the LANL database, and we assume a read length of 150 bp. Recall that the average genome length for this dataset is 8970 bp, so each sequence contains about 60 reads' worth of data, on average. For each sequence, we select 60 random positions, take the subsequence of length 150 bp starting at each position, and concatenate these 60 subsequences to form a new sequence – in this way, we simulate the process of a DNA sequencer. Then, we repeat the same 10-fold cross-validation at $k = 6$ and with the

linear SVM classifier as before, but with this new set of "stitched-together" sequences. We obtain an accuracy of 96.46% (compared to an accuracy of 96.49% with the original sequences), demonstrating the applicability of our method to unassembled read data. We also rerun the same experiment but using fewer samples per sequence, with the results shown in Figure 3.6. As can be seen, fewer samples give lower accuracy but good performance is still achieved even with a low degree of coverage of the original sequence.

Figure 3.6: **Classification accuracy scores for the HIV-1 simulated NGS read experiment, with different numbers of samples per sequence (each sample of length 150 bp).**



Because of the exponential growth of sequence databases, modern bioinformatics tools increasingly must be capable of handling NGS sequence data and must be scalable enough to manage huge sets of data. As well, researchers often demand the privacy, security, and reproducibility characteristics an open-source, standalone, offline tool such as ours provides. However, there remain several areas for future work. Although our tool matches or exceeds the classification speed of the competing state-of-the-art, performance optimization was not a focus of this study and we believe there is room to substantially

improve running time even further. Similarly, although we match or exceed the classification accuracy of the competing state-of-the-art, different modern machine learning methods such as GeneVec [7] or deep neural networks may permit us to achieve even higher accuracy on challenging datasets. As well, given the rapid rate of mutation of many viruses, it would be highly useful for our tool to be capable of giving a result of 'no match' with its training data. Each of these possibilities could make our method and software even more useful in the future.

Chapter 4

Taxonomic classification of genomic sequences: demonstrating generality

4.1 Taxonomic classification

We have demonstrated that our method is highly successful in viral subtyping, particularly of HIV-1, but our results in this study will demonstrate its potential to be applicable to a variety of different genomic sequence classification tasks. For our next set of experiments, we turn our attention to the problem of taxonomic classification. For this task, we wish to assign a sequence to the phylogenetic group of the organism it belongs to – for example, we may wish to determine whether a sequence from an unknown vertebrate belongs to one of the classes of subphylum *Vertebrata*, such as *Amphibia*, *Aves*, *Mammalia*, or *Reptilia*. A solution to this problem may help researchers determine the taxonomic classification of a newly discovered organism, or to resolve controversial phylogenetic assignments.

We wish to demonstrate that our method is able to solve the taxonomic classification problem across as much of the spectrum of genomic diversity as possible. For this reason, we evaluate our method on a total of 28 datasets related to taxonomic classification, including whole genomes from a variety of sources: mitochondrial, nuclear, plastid, plasmid, and viral; and taxonomic groupings at every level from kingdom down to family.

We obtain mitochondrial sequences from the National Center for Biotechnology Information (NCBI) RefSeq sequence database release version 81, plastid sequences from NCBI RefSeq release version 82, plasmid sequences from NCBI RefSeq release version 83, nucleoid genomes from the NCBI genomes browser at https://www.ncbi.nlm.nih.gov/genome/browse/ with a level of 'complete' or 'chromosome', and viral genomes from the NCBI Nucleotide database with query `"txid10239"[Organism:exp] AND ("complete genome"[Title] OR "complete sequence"[Title]) NOT "miRNA"[Title] NOT "long terminal repeat"[Title] NOT "ltr"[Title] NOT "contig"[Title] NOT "spacer"[Title] NOT "pseudogene"[Title] NOT "genes"[Title] NOT "gene"[Title] NOT "segment"[Title] NOT "partial"[Title] NOT "cds"[Title] NOT "except"[Title] NOT "region"[Title] NOT "incomplete"[Title]`. In the case of nucleoid genomes with multiple chromosomes, we concatenate all chromosomes to produce a single genome sequence. We select 28 subsets of these sequences as shown in Table 4.1; Table 4.2 additionally shows the number of sequences and number of classes for each dataset. As discussed in the previous chapter, machine learning models are unable to 'learn' unless given a sufficient quantity of training data, and to this end every dataset was constructed so that every class had a minimum of 10 elements.

Since we use cross-validation to measure accuracy on these datasets, it

is necessary to ensure no duplication exists between the training and testing partitions. This may occur if a dataset has many sequences of the same species – in this case, we may be testing our model on sequences highly similar to sequences used for training, which would unfairly inflate accuracy. We use the RefSeq sequence databases to avoid this case, because RefSeq databases are curated to contain at most one representative sequence for any species.

Table 4.1: **Descriptions of datasets used for taxonomic classification experiments on whole genomes.**

Name	Description
genomes-nuclear/5kingdoms	Nucleoid genomes, split into 5 of the 6 kingdoms: animals, archaea, bacteria, fungi, and plants
genomes-nuclear/archaea	Archaeal nucleoid genomes, split into 3 phyla
genomes-nuclear/bacteria	Bacterial nucleoid genomes, split into 4 phyla
genomes-nuclear/proteobacteria	Proteobacterial nucleoid genomes, split into 5 classes
genomes-nuclear/fungi	Fungal nucleoid genomes, split into 3 phyla or subphyla
genomes-nuclear/plants	Plant nucleoid genomes, split into 2 clades
genomes-nuclear/vertebrates	Vertebrate genomes into birds, fish, and mammals
mtdna/amphibians	Amphibian mitochondrial genomes, split into 3 orders
mtdna/fungi	Fungal mitochondrial genomes, split into 3 phyla or subphyla
mtdna/ insects-mammals-amphibians	Animal mitochondrial genomes, split into insects, mammals, and amphibians
mtdna/insects	Insect mitochondrial genomes, split into 7 orders or superorders
mtdna/mammals	Mammal mitochondrial genomes, split into 8 orders or superorders
mtdna/ plants-animals-fungi-protists	Eukaryote mitochondrial genomes, split into plants, animals, fungi, and protists
mtdna/plants	Plant mitochondrial genomes, split into 2 clades
mtdna/primates	Primate mitochondrial genomes, split into 2 suborders
mtdna/protists	Protist mitochondrial genomes, split into 3 superphyla
mtdna/vertebrates	Vertebrate mitochondrial genomes, split into amphibians, birds, fish, mammals, and reptiles
plasmids/bacteria	Bacterial plasmid genomes, split into into 4 phyla
plasmids/proteobacteria	Protobacterial plasmid genomes, split into into 3 classes
plastids/plants	Plant plastid genomes, split into into 5 clades
plastids/protists	Protist plastid genomes, split into into 3 superphyla
viruses/dsDNA	Genomes from double-stranded DNA virus with no RNA stage, split into 6 families
viruses/groups	Viral genomes, split into 6 groups from the Baltimore virus classification
viruses/retrotranscribing	Retrotranscribing virus genomes, split into 6 families
viruses/satellites	Satellite virus genomes, split into 6 families
viruses/ssDNA	ssDNA virus genomes, split into 4 families
viruses/ssRNAnegative	ssRNA negative-strand virus genomes, split into 4 families
viruses/ssRNApositive	ssRNA positive-strand virus genomes, split into 7 families

Table 4.2: **Statistics for datasets used for taxonomic classification experiments on whole genomes.**

Name	# of classes	# of sequences
genomes-nuclear/5kingdoms	5	3362
genomes-nuclear/archaea	3	209
genomes-nuclear/bacteria	4	2500
genomes-nuclear/proteobacteria	5	1350
genomes-nuclear/fungi	3	71
genomes-nuclear/plants	2	66
genomes-nuclear/vertebrates	3	57
mtdna/amphibians	3	290
mtdna/fungi	3	226
mtdna/insects-mammals-amphibians	3	2170
mtdna/insects	7	898
mtdna/mammals	8	830
mtdna/plants-animals-fungi-protists	4	7385
mtdna/plants	2	254
mtdna/primates	2	148
mtdna/protists	3	160
mtdna/vertebrates	5	4327
plasmids/bacteria	4	8664
plasmids/proteobacteria	3	4691
plastids/plants	5	1208
plastids/protists	3	126
viruses/dsDNA	6	6630
viruses/groups	6	50112
viruses/retrotranscribing	2	9946
viruses/satellites	2	1574
viruses/ssDNA	4	7927
viruses/ssRNAnegative	4	3460
viruses/ssRNApositive	7	16650

Each of these datasets and others found in this chapter are available online at https://github.com/stephensolis/kameris-experiments; on the same page can also be found step-by-step instructions for the reproduction of every experiment presented here.

Again following the example of the previous chapter, we perform 10-fold cross-validation on each of these datasets, at $k = 6$ and with the linear SVM

classifier, in order to measure performance. In all cases, we obtain extremely high accuracy scores, as shown in Table 4.3.

Table 4.3: **Classification accuracy results for taxonomic classification experiments.**

Name	Classification accuracy
genomes-nuclear/5kingdoms	97.08%
genomes-nuclear/archaea	97.14%
genomes-nuclear/bacteria	98.72%
genomes-nuclear/proteobacteria	96.96%
genomes-nuclear/fungi	94.46%
genomes-nuclear/plants	92.86%
genomes-nuclear/vertebrates	98.33%
mtdna/amphibians	100%
mtdna/fungi	96.46%
mtdna/insects-mammals-amphibians	100%
mtdna/insects	98.89%
mtdna/mammals	99.76%
mtdna/plants-animals-fungi-protists	99.54%
mtdna/plants	96.45%
mtdna/primates	100%
mtdna/protists	98.75%
mtdna/vertebrates	99.95%
plasmids/bacteria	97.67%
plasmids/proteobacteria	97.17%
plastids/plants	99.42%
plastids/protists	96.92%
viruses/dsDNA	99.50%
viruses/groups	96.37%
viruses/retrotranscribing	100%
viruses/satellites	100%
viruses/ssDNA	98.74%
viruses/ssRNAnegative	99.55%
viruses/ssRNApositive	98.54%

However, these accuracy scores could be even higher. Double-stranded DNA molecules can have either their sense or antisense strand sequenced, and by convention, all sequences in the NCBI RefSeq database should be of the sense strand [133]. k-mer count vectors are clearly different for a sequence

and its reverse complement, so if a classifier were trained on sequences of one sense and tested on a sequence of the opposite sense, one would not expect the classifier to be capable of making an accurate prediction. In fact, we have evidence this occurs: in the `mtdna/vertebrates` dataset, we have an accuracy score of 99.95%, and only 3 sequences were misclassified, namely *Serinus canaria* (NCBI ID NC_023375.1), *Channa micropeltes* (NC_030542.1), and *Sinibotia reevesae* (NC_030322.1). For each of these sequences, we find another sequence in the dataset from the same genus, and we compute the Manhattan distance from its k-mer proportion vector to both the original sequence and its reverse complement, and find a much lower distance to the reverse complement for all 3 cases. This suggests those genomes were mistakenly included in their antisense form rather than the sense form like the rest of the dataset. When we re-run the experiment with those sequences replaced with their reverse complement, we obtain a classification accuracy of 100%.

More generally, we observe that our method does not gracefully handle the case of a reverse-complemented sequence, which can be an issue for some datasets. Some other k-mer counting-based solutions solve the problem by concatenating every sequence with its reverse complement before computing counts, but more research is needed to determine whether accuracy could be impacted by doing so.

We may also generate MoDMap plots for these datasets, which reveals one of the limitations of MoDMaps. In the previous chapter, we propose MoDMaps as an unsupervised data exploration tool, and we demonstrate cases where they are successful in showing known relationships between viral subtypes. Similarly, for some of these datasets, MoDMaps do a good job in depicting class relationships, for instance with the primate suborders in Figure 4.1. However, for other datasets, the MoDMap does not show much structure,

for instance with the virus groups in Figure 4.2, even though classification accuracy is over 96% for those data. It is important to note that a 'messy' MoDMap does not imply poor classification performance.

Figure 4.1: **MoDMap of whole primate mitochondrial genomes, split into suborders, at $k = 6$.**

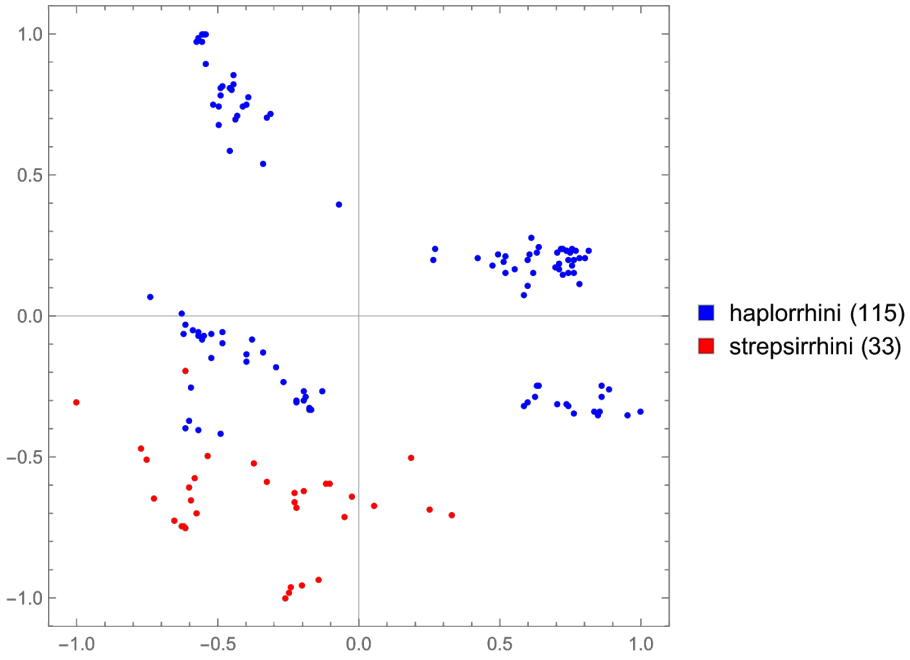
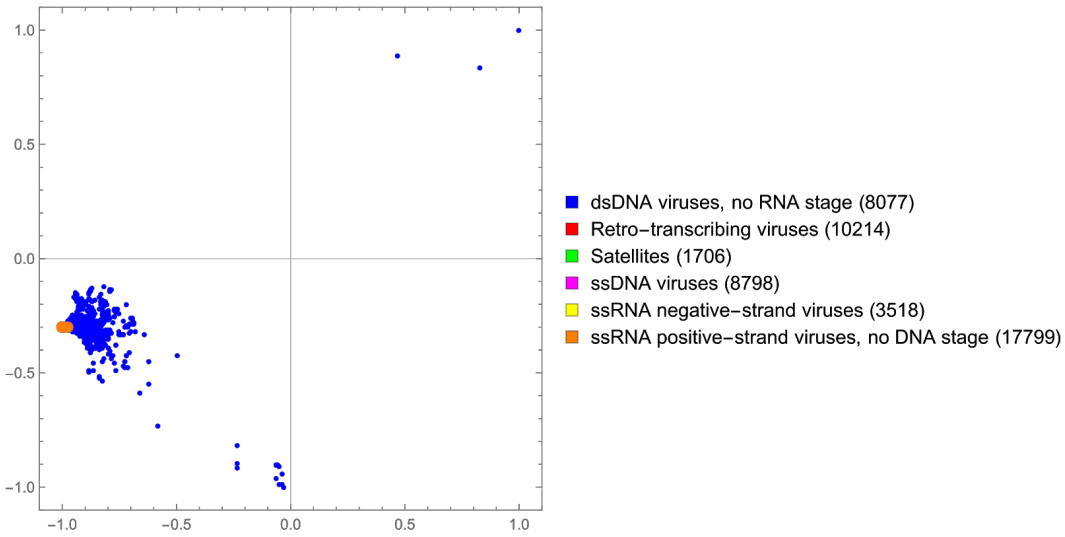


Figure 4.2: **MoDMap of whole viral genomes, split into groups, at $k = 6$.**



Ours is not the only k-mer-based software capable of performing taxonomic classification, and a different one is Logic Alignment Free (LAF) of Weitschek *et al.* [167]. Essentially, LAF works by generating a list of small if-then rules capable of determining whether a sequence belongs to a class of data – an example of such a rule would be $(f(ACGT) > 0.15) \wedge (f(GGCT) < 0.6)$. As part of their work, the LAF authors benchmark their algorithm on 9 partitionings of a dataset of 1964 whole bacterial genomes, so we perform a head-to-head comparison using Kameris. The class splitting performed by the LAF authors was done by taxonomic level, and they additionally produce 'filtered' and 'not filtered' versions of the same datasets where the 'filtered' versions omit any genomes belonging to species with fewer than 9 other specimens in the dataset: there are 413 genomes in the filtered sets. We use $k = 4$ to match the settings used with LAF and use the linear SVM classifier, and we obtain accuracy scores listed in Table 4.4. As can be seen, we outperform LAF in 8 of the 9 datasets, in some cases by as much as 5%. This demonstrates Kameris is competitive with the state-of-the-art in the task of taxonomic classification.

Table 4.4: **Classification accuracy and dataset statistics of Kameris vs. LAF on datasets of whole bacterial genomes.**

Dataset	**Classification accuracy**	
	Kameris	**LAF**
25 species, filtered	99.76%	97.61%
21 genera, filtered	100%	98.79%
14 orders, filtered	100%	99.27%
9 classes, filtered	100%	98.79%
6 phyla, filtered	99.75%	98.78%
590 genera, not filtered	71.28%	73.04%
120 orders, not filtered	87.02%	85.68%
57 classes, not filtered	90.17%	89.10%
36 phyla, not filtered	91.17%	86.08%

4.2 Transcriptome data

In the previous chapter, we provide some evidence to suggest KAMERIS does not require contiguous, assembled sequence data, and it works almost equally well when 'stitching together' short sequence fragments. This capability is highly applicable to dealing with reads from next-generation sequencing (NGS) technologies without having the requirement to first assemble the sequences. To provide further evidence, we perform some more 'stitching together' experiments, this time with datasets of messenger RNA (mRNA) transcriptome data. As opposed to a full genome, the transcriptome comprises only that part of an organism's genetic information transcribed to RNA by RNA polymerase.

Specifically, we consider two sources of data: first, the data from the Marine Microbial Eukaryotic Transcriptome Sequencing Project (MMETSP) [90]. This is a collection of 678 transcriptomes of abundant and ecologically significant microbial eukaryotes from the oceans, assembled at the gene level. A key goal of the MMETSP project was to build a dataset representative of the organisms in a typical environmental sample. Within this dataset, every gene is given in two forms: 'nt' which is the whole mRNA transcript, and 'cds' which is only the portion of the transcript translated by a ribosome. As well, we work with the data from the 1000 Plants (1KPlants) project [109]. Despite the name, this is a collection of 1314 transcriptomes from species of kingdom *Plantae*. Assembly is again done to the gene level, and sequences are of whole mRNA transcripts.

We again want to perform taxonomic classification, so we begin by partitioning the MMETSP data 6 ways, into groups given by different taxonomic levels – from high-level to low: 6 superphyla, 10 phyla, 15 classes, 22 orders, 19

families, and 18 genera. For each transcriptome, we construct a single sample sequence by randomly selecting genes and concatenating them until the total sample length is greater than some variable threshold. We again perform 10-fold cross-validation with $k = 6$ and the linear SVM classifier, with the results shown in Table 4.5. We observe that even 50 kbp of sequence data suffice to accurately classify transcriptomes on taxonomic groups, with more data giving more accurate results. Further, we see that the whole-transcript ('nt') dataset gives generally lower accuracy than the coding-region ('cds') dataset, suggesting that the non-coding region of a mRNA transcript is in fact not useful for taxonomic classification.

Table 4.5: **Classification accuracy results for classification of the MMETSP transcriptomes, with different lengths of random samples.** 'nt' denotes whole-transcript data and 'cds' denotes coding-region-only data.

Dataset	**Classification accuracy**		
	10 kbp sample	**50 kbp sample**	**500 kbp sample**
6 superphyla, 'nt'	79.87%	88.57%	92.59%
6 superphyla, 'cds'	82.93%	90.98%	93.23%
10 phyla, 'nt'	81.17%	93.54%	93.73%
10 phyla, 'cds'	84.58%	92.47%	95.88%
15 classes, 'nt'	78.20%	90.86%	93.59%
15 classes, 'cds'	82.11%	92.03%	94.56%
22 orders, 'nt'	67.27%	89.58%	93.65%
22 orders, 'cds'	72.57%	90.35%	93.67%
19 families, 'nt'	76.53%	93.08%	95.48%
19 families, 'cds'	78.30%	95.17%	95.18%
18 genera, 'nt'	69.03%	93.33%	97.19%
18 genera, 'cds'	75.02%	92.63%	96.82%

We repeat the exact same experiment again but this time after partitioning the 1KPlants data 3 ways, also into groups given by different taxonomic levels – from high-level to low: 15 clades, 27 orders, and 28 families, with the results shown in Table 4.6. Interestingly, we note that the difference in classi-

fication accuracy between amounts of sampling is much more pronounced for the 1KPlants data versus the MMETSP data – since land plants tend to have fairly large genomes, this result may suggest that their genomes have a lower signal-to-noise ratio in terms of taxonomic classification.

Table 4.6: **Classification accuracy results for taxonomic classification of the 1,000 Plants Project transcriptomes, with different lengths of random samples.**

Dataset	**Classification accuracy**		
	10 kbp samples	**50 kbp samples**	**500 kbp samples**
15 clades	61.04%	80.59%	93.56%
27 orders	54.69%	75.00%	89.74%
28 families	58.47%	85.89%	96.98%

As mentioned, supervised classifiers typically perform better if given more training data. To test this, we repeat the same experiment, with the whole-transcript datasets and 50 kbp samples, but take multiple samples per transcriptome. Since this would increase the number of points per class, this may make it easier for the classifier to learn the classes, increasing accuracy. In order to avoid concerns of overfitting, we ensure all samples from a particular organism are placed in the same cross-validation group. Results can be seen in Table 4.7. We observe that the accuracy obtained by 5 samples of length 50 kbp is about the same as, but slightly lower than, that of 1 sample of length 500 kbp. Further, we do not see an improvement in performance beyond 5 samples per transcriptome. This suggests that, at least for this dataset, training a model on multiple samples per organism does not improve accuracy compared to a single longer sample.

Table 4.7: **Classification accuracy results for taxonomic classification of MMETSP transcriptomes, with samples of length 50 kbp and different numbers of samples per transcriptome.**

Taxonomic level	**Classification accuracy**		
	1 sample	**5 samples**	**10 samples**
6 superphyla	88.57%	90.82%	90.95%
10 phyla	93.54%	94.29%	93.93%
15 classes	90.86%	93.66%	93.23%
22 orders	89.58%	94.06%	92.81%
19 families	93.08%	95.05%	95.36%
18 genera	93.33%	96.62%	96.62%

Overall, these results are particularly surprising since every sequence given to the classifier is composed of sequence data randomly sampled from different parts of each genome. Using an alignment-based technique here would be impossible, since there is no common data to be aligned – each sample is of a different set of genes. The fact that classification performance is high suggests that k-mer bias patterns persist and are preserved across the entire transcriptome, with a strong enough signal to allow accurate taxonomic classification of samples as short as 10 kbp.

There are other interesting transcriptome datasets which could be explored – for instance, NCBI GenBank has a collection of tens of thousands of vertebrate transcriptomes. Also, in order to add still more support to the idea of randomly sampling genomes, it may be interesting to reproduce the taxonomic classification experiments from the previous section while randomly sampling and 'stitching together' all genomes in the same way as was done with the HIV-1 sequences. Even more interesting would be to determine classification accuracy for datasets taken from the NCBI Sequence Read Archive (SRA), which is a collection of real NGS read data. In this work, we intentionally avoided using real read data because there are some important data cleanup steps needed before such data would be usable – low-quality, low-

entropy (for example, ATATAT...), and duplicated (multiple reads of almost exactly the same region) reads would need to be filtered out to avoid adding noise to the classifier. By using only data which has been at least partially assembled, we allow the assembly pipeline to perform these cleanup steps for us. In the future, we could add functionality to KAMERIS to support read filtering directly.

4.3 Intra-species classification

So far, we have demonstrated success on the tasks of viral subtyping and taxonomic classification, with very high accuracy. We have demonstrated we can even classify accurately after randomly sampling the genomes or sequences being classified. Here, we present further results on classifying sets of sequences of the same species, and demonstrate how KAMERIS still performs very well.

First, we consider human mitochondrial DNA. In humans, mitochondrial DNA does not undergo genetic recombination, and is inherited solely through the mother's side. Thus, mitochondrial DNA can be used to identify maternal lineage, and the human matrilinear line has been organized into haplogroups, identified by a letter sometimes followed by letters and numbers. Every haplogroup is defined by a panel of specific single-nucleotide polymorphisms (SNPs) at specific positions of the mitochondrial genome – typically, these panels have about 20 positions. Although SNP panels are used for haplogroup determination, they are not necessarily the only mutations between genomes of different haplogroups. Indeed, since a single point mutation may only change the proportions of at most $2k-1$ surrounding k-mers, it is unlikely that such a small number of SNPs would, on its own, result in enough information for use in classification. Our method only looks at substrings of length

k, so we capture no information about the position of a mutation beyond its surrounding window of size $k-1$.

We consider the set of whole human mitochondrial genomes, partitioned by haplogroup from the MitoMap project [107]. Since this dataset has just under 28,000 sequences, and in order to avoid issues with class size imbalance, we omit any haplogroup classes with fewer than 100 examples. This gives 66 different haplogroup classes and 23 classes of top-level haplogroups (that is, considering only the first letter of the haplogroup name). As before, we perform a 10-fold cross-validation with $k = 6$ and the linear SVM classifier, and we obtain a classification accuracy of 98.43% with the haplogroup classes and 99.55% with the top-level haplogroups. This result demonstrates that, although haplogroups are defined by a few tens of SNPs, they must also exhibit overall k-mer proportion biases, which our classifier is able to recognize and use. As mentioned, if overall k-mer biases would not exist, we would have insufficient information to accurately distinguish haplogroups with our method. More work is needed to further investigate such biases, and to determine a biological explanation for them.

Some diseases are similar to haplogroups in that they are linked to a specific set of SNPs. For example, congenital lactose intolerance in humans is linked to a single point mutation in chromosome 2. In this work, we do not explore the possibility of using Kameris to diagnose genetic disease, but given this result, such a study may prove to be fruitful.

In order to further explore the idea of 'unexpected' k-mer proportion biases allowing better classification, we consider again the set of influenza virus genomes from Chapter , but this time considering genomic regions rather than whole genomes. Influenza virus genomes are divided into 8 distinct segments: in order, these are PB2, PB1, PA, HA, NP, NA, MP, and NS. Subtypes of

influenza are determined by a code 'H?N?' where the '?' are numbers, and this code represents a classification based on the HA and NA genomic regions. The remaining regions should not be relevant for the purpose of influenza virus subtyping, since subtype assignment is not done using those regions at all. However, in the same way as our result with human mitochondrial DNA haplogroups, we predict training a subtype classifier using regions other than HA and NA may in fact work.

We test this by taking the set of influenza genomes from Chapter 2, and for every genome, we extract each of the 8 segments in turn, and split the set of genomes both into subtypes (H?N?), HA-only subtype (H?), and NA-only subtype (N?). We perform a 10-fold cross-validation on each set, with results in Table 4.8 – for comparison, accuracy for the full subtype using the full genome was 96.68%. As can be seen, we confirm our hypothesis that the 'irrelevant' regions in fact do have the ability to predict subtypes. Although the accuracy scores for those regions are lower than those for the HA and NA regions, they are fairly significant, and in the same way as the human haplogroup experiment, this result suggests that there are some influenza-genome-wide k-mer proportion biases which the classifier is able to recognize and use.

Table 4.8: **Classification accuracy results for subtyping of segments of influenza genomes, for full subtypes (H?N?), HA subtype (H?), and NA subtype (N?).**

Segment	Classification accuracy		
	Full subtype	**HA subtype**	**NA subtype**
PB2	79.57%	78.16%	77.20%
PB1	79.90%	78.41%	77.53%
PA	79.32%	78.44%	76.73%
HA	**90.51%**	**97.45%**	**87.62%**
NP	78.82%	77.92%	76.06%
NA	**89.83%**	**87.31%**	**97.57%**
MP	78.19%	78.14%	76.17%
NS	77.52%	77.58%	75.67%

4.4 Conclusions

In this chapter, we demonstrate Kameris is highly successful in classifying genomic sequences by taxonomic group. We show very high classification accuracy scores for 28 different datasets composed of whole mitochondrial, nuclear, plastid, plasmid, and viral genomes for taxonomic groupings at every level from kingdom down to genus. This is in contrast with previous studies in this space, which typically test methods with one or a small number of datasets, typically of one type of genome and one or few taxonomic levels. We perform a head-to-head comparison with Logic Alignment Free (LAF), a competing tool, and find we exceed its classification accuracy on several datasets of whole bacterial genomes.

We further classify with high accuracy randomly sampled transcriptomes of marine microbes and plants by taxonomic group, human mitochondrial genomes into haplogroups, and partial influenza genomes into subtypes. These results all suggest the presence of genome-wide k-mer proportion biases, which certainly deserves further study.

Chapter 5

Conclusions and Future Work

In this work, we present a remarkably simple, supervised, alignment-free method for sequence classification based on k-mer counting, and we implement the method in a fully standalone, easy to use, open-source software package called Kameris.

We comprehensively demonstrate its general applicability and flexibility by computing its accuracy in the subtyping of HIV-1, dengue, influenza A, hepatitis B, and hepatitis C virus genomes in Chapter 3; the taxonomic classification of whole mitochondrial, nuclear, plastid, plasmid, and viral genomes in Section 4.1, and sampled marine eukaryote and plant transcriptomes in Section 4.2, into taxonomic groupings at every level from kingdom down to genus; and the determination of human haplogroups from mitochondrial genomes in Section 4.3. We show how accurate classification remains possible even when using only the *pol* gene region of the HIV-1 genome, or only a single segment of the influenza virus genome. We perform head-to-head comparisons with competing state-of-the-art software in the tasks of HIV-1 subtyping and taxonomic classification of whole bacterial genomes, and show that we match or exceed all competitors in accuracy and speed. Further, we demonstrate

the applicability of our method to NGS read data by showing it is accurate even when using randomly sampled HIV-1 genomes and marine eukaryote and plant transcriptomes, and study the amount of sequence data needed to obtain accurate classification.

However, we identify a number of important limitations. As opposed to, for example, sequence alignment, our classifier algorithms generally act as a black box and do not provide an accessible explanation as to why a sequence is classified in a certain way. As well, the classifier algorithms we use require a sizable, clean set of training data. Further, we have no ability to provide a 'no match' result in case the given sequence does not match any training set classes.

There exist some means by which we may try to gain some insights into classification rationale, so as to peek inside the 'black box'. One technique is recursive feature elimination (RFE) [58] – essentially, it works by training a classifier capable of assigning weights to features, eliminating the least-weighted features, and repeating until there are no more features. In this way, it is able to estimate the relative 'importance' of each feature, and thus identify the specific k-mers which are particularly relevant for a particular classification task. We apply the RFE algorithm to the `mtdna/vertebrates` taxonomic classification dataset from the previous chapter, using the linear SVM classifier and $k = 5$, and without performing dimensionality reduction on the k-mer vectors. $k = 5$ was chosen for ease of plotting. We show the results by wrapping the 1024-element vector into a 32x32 square, as shown in Figure 5.2 – the elements of the square are ordered as in Chaos Game Representation plots, described in [76]. More specifically, the order is recursive by quadrant of the square, as shown in Figure 5.1. For example, the small square at the top left of the figure represents the 5-mer `CCCCC`. Each square in the

figure represents the importance of a particular k-mer, with values being relative importance, that is, two squares with value 200 and 400 would indicate the first corresponding k-mer being half as important as the other. We find some k-mers identified as important but more research is needed to determine whether the k-mers identified as important have any biological relevance. Feature importance is not necessarily limited to single k-mers, but correlations or interdependence between k-mers may be relevant as well and also deserves study.

Figure 5.1: **Diagram of k-mers in a Chaos Game Representation plot.**

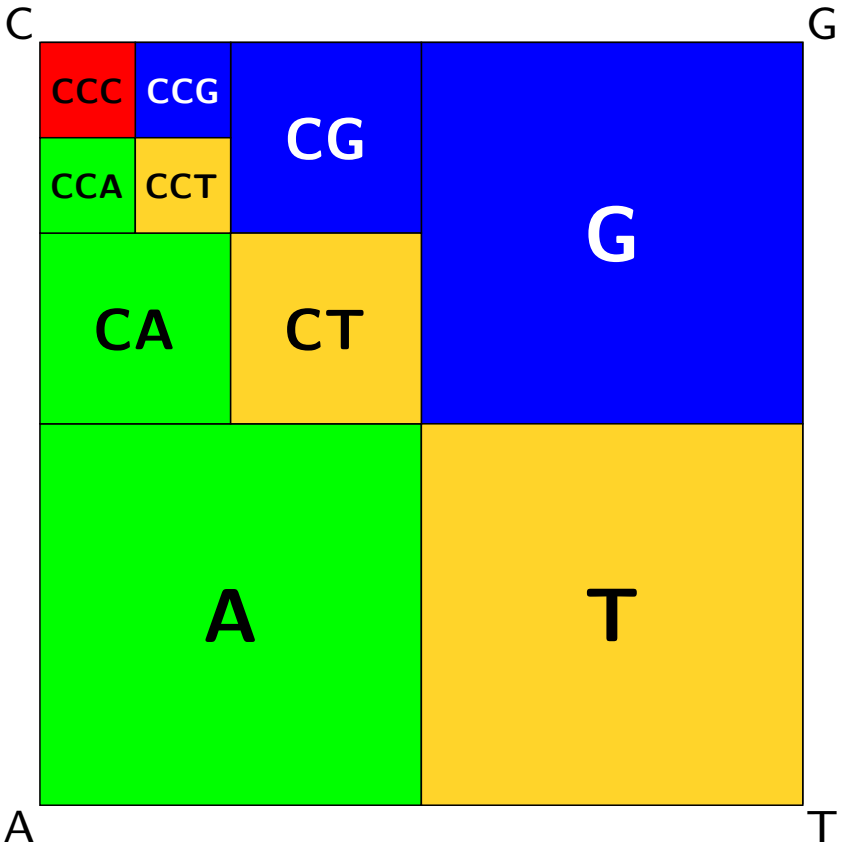
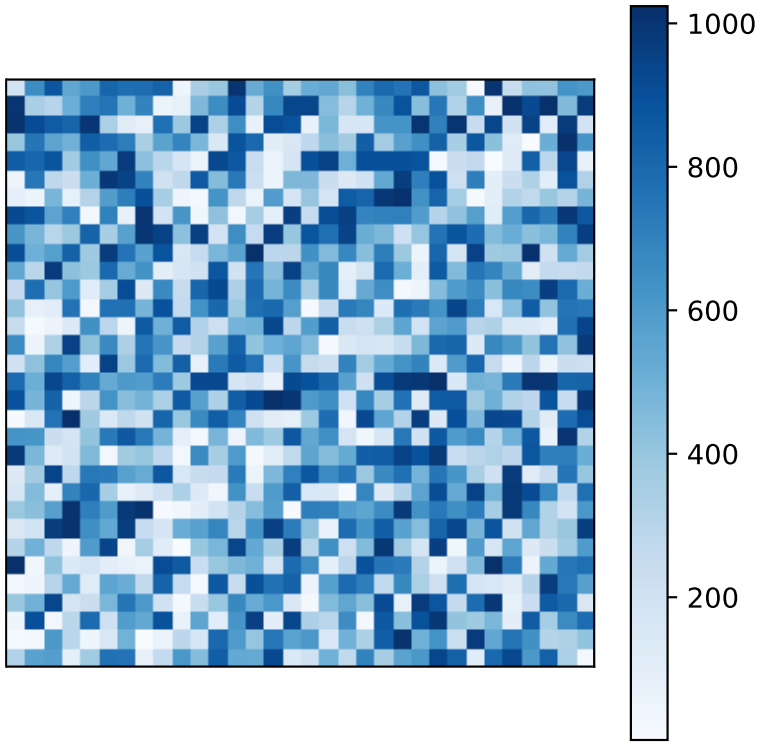


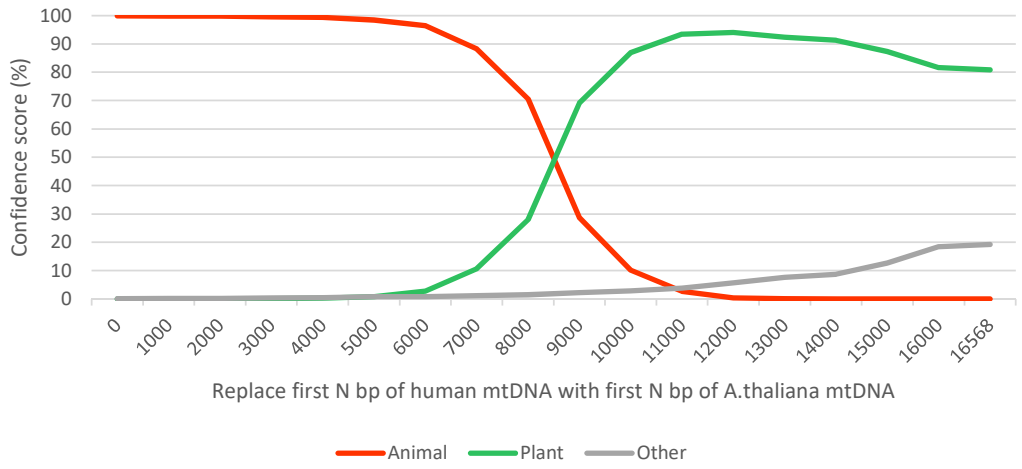
Figure 5.2: **Plot of RFE-determined k-mer importance values, for `mtdna/vertebrates` using the linear SVM classifier at $k = 5$.** Each square in the figure represents the importance of a particular k-mer, with values being relative importance (two squares with value 200 and 400 would indicate the first corresponding k-mer being half as important as the second). The 1024-element vector is wrapped into a 32x32 square for ease of display.



As well, although we do not support a 'no match' result, it is interesting to consider cases where data is 'between' two classes in some sense. This has particular relevance to synthetic biology, where sequence data may be mixed from multiple species. For this, we require a classifier capable of outputting confidence scores, so we use the logistic regression classifier. We begin with the human mitochondrial genome, and we progressively replace the first N base pairs of the genome with the first N base pairs of the *A.thaliana* mitochondrial genome. We train a logistic regression classifier on the `mtdna/plants-animals-fungi-protists` dataset (with plant, animal, fungi, and protist classes), and show the confidence scores for the classes in

Figure 5.3. We see that the classifier is indeed able to infer a mix of classes, and we see the predictions cross over exactly when replacing half of the sequence. In a similar way, it is interesting to consider the resilience of our method to mutational events – specifically, how many point mutations may be introduced to a sequence before our model loses the ability to accurately classify it; this has implications for the real-world use of our method and deserves further research.

Figure 5.3: **Plot of confidence score of different classes for sequences composed of different proportions of the human and *A.thaliana* mitochondrial genomes, using a classifier trained on plants, animals, fungi, and protists.** Sequences are constructed by replacing the first N base pairs of the human mitochondrial genome with the first N base pairs of the *A.thaliana* mitochondrial genome, with N given on the x-axis. 'Other' means any class other than animal or plant.



There are a few other potential avenues of research. Given the demonstrated generality of this method, it may be interesting to try an application to protein classification – however this would require careful attention to the construction of k-mer vectors, since using the protein alphabet directly would result in an exponential growth of vector length, which may make training intractable. It would also be interesting to look in more detail at the datasets,

such as plant nuclear DNA, which performed more poorly than others: there may be some interesting biological reasons for it. Also, there were some assumptions we made without quantitative evidence, which may not have been justified: we can assess how small training classes may be before classification becomes impossible rather than arbitrarily selecting minimum class sizes; whether our selection of $k = 6$ remains optimal for datasets other than HIV-1 genomes, and if not, how the optimal value of k changes; whether raw reads may be classified accurately without performing low-quality, low-entropy, and redundancy filtering; and whether we can use our method to diagnose genetic diseases. Regarding genetic diseases specifically, it is known that some diseases, such as some forms of cancer, cause systemic mutations across a whole genomic region – our method may be able to use this to make accurate predictions. As well, since computational performance was not a focus of this study, relatively large speed improvements may be easy to obtain. Any of directions could help make Kameris even more general and useful.

Bibliography

[1] Takashi Abe et al. "A novel bioinformatics tool for phylogenetic classification of genomic sequence fragments derived from mixed genomes of uncultured environmental microbes". In: *Polar Research* 20 (2006), pp. 103–112.

[2] Takashi Abe et al. "Novel phylogenetic studies of genomic sequence fragments derived from uncultured microbe mixtures in environmental and clinical samples". In: *DNA Research* 12.5 (2005), pp. 281–290.

[3] Emmanuel Adetiba, Oludayo O. Olugbara, and Tunmike B. Taiwo. "Identification of pathogenic viruses using genomic cepstral coefficients with radial basis function neural network". In: *Proceedings of the 7th World Congress on Nature and Biologically Inspired Computing (NaBIC2015).* Ed. by Nelishia Pillay et al. Springer International Publishing, 2016, pp. 281–291.

[4] Aydin Albayrak, Hasan H Otu, and Ugur O Sezerman. "Clustering of protein families into functional subtypes using Relative Complexity Measure with reduced amino acid alphabets". In: *BMC Bioinformatics* 11 (2010), p. 428.

[5] Naomi S. Altman. “An introduction to kernel and nearest-neighbor nonparametric regression”. In: *The American Statistician* 46.3 (1992), pp. 175–185.

[6] Stephen F. Altschul et al. “Basic local alignment search tool”. In: *Journal of Molecular Biology* 215.3 (1990), pp. 403–410.

[7] Ehsaneddin Asgari and Mohammad R. K. Mofrad. “Continuous Distributed Representation of Biological Sequences for Deep Proteomics and Genomics”. In: *PLoS One* 10.11 (2015), pp. 1–15.

[8] Nazneen Aziz et al. “College of American Pathologists’ laboratory standards for next-generation sequencing clinical tests”. In: *Archives of Pathology and Laboratory Medicine* 139.4 (2014), pp. 481–493.

[9] Yasin Bakış et al. “Testing robustness of relative complexity measure method constructing robust phylogenetic trees for Galanthus L. using the relative complexity measure”. In: *BMC Bioinformatics* 14 (2013), p. 20.

[10] Dhundy R Bastola et al. “Utilization of the relative complexity measure to construct a phylogenetic tree for fungi”. In: *Mycological Research* 108.2 (2004), pp. 117–125.

[11] Ernest Beutler et al. “Evolution of the genome and the genetic code: selection at the dinucleotide level by methylation and polyribonucleotide cleavage”. In: *Proceedings of the National Academy of Sciences of the United States of America* 86.1 (1989), pp. 192–196.

[12] Ashok S Bhagwat and Michael McClelland. “DNA mismatch correction by Very Short Patch repair may have altered the abundance of oligonucleotides in the E. coli genome”. In: *Nucleic Acids Research* 20.7 (1992), pp. 1663–1668.

[13] Christopher Bishop. “Pattern recognition and machine learning”. In: Springer-Verlag New York, 2006. Chap. 4.3.4: Multiclass logistic regression, pp. 209–210.

[14] B Edwin Blaisdell. “A measure of the similarity of sets of sequences not requiring sequence alignment”. In: *Proceedings of the National Academy of Sciences of the United States of America* 83.14 (1986), pp. 5155–5159.

[15] B Edwin Blaisdell. “Effectiveness of measures requiring and not requiring prior sequence alignment for estimating the dissimilarity of natural sequences”. In: *Journal of Molecular Evolution* 29.6 (1989), pp. 526–537.

[16] Jon Bohlin. “Genomic signatures in microbes – properties and applications”. In: *The Scientific World Journal* 11 (2011), pp. 715–725.

[17] Jon Bohlin and Eystein Skjerve. “Examination of genome homogeneity in prokaryotes using genomic signatures”. In: *PLoS One* 4.12 (2009).

[18] Oliver Bonham-Carter, Joe Steele, and Dhundy Bastola. “Alignment-free genetic sequence comparisons: a review of recent approaches by word analysis”. In: *Briefings in Bioinformatics* 15.6 (2013), pp. 890–905.

[19] Ingwer Borg and Patrick Groenen. *Modern Multidimensional Scaling: Theory and Applications*. 2nd ed. Springer, 2005.

[20] Leo Breiman. “Random forests”. In: *Machine Learning* 45.1 (2001), pp. 5–32.

[21] Leo Breiman et al. *Classification and regression trees*. Wadsworth Statistics/Probability. Chapman and Hall, 1984.

[22] Chris Burge, Allan M Campbell, and Samuel Karlin. "Over- and under-representation of short oligonucleotides in DNA sequences". In: *Proceedings of the National Academy of Sciences of the United States of America* 89.4 (1992), pp. 1358–1362.

[23] Allan M Campbell, Jan Mrázek, and Samuel Karlin. "Genome signature comparisons among prokaryote, plasmid, and mitochondrial DNA". In: *Proceedings of the National Academy of Sciences of the United States of America* 96.16 (1999), pp. 9184–9189.

[24] Tony F Chan, Gene Howard Golub, and Randall J LeVeque. "Updating formulae and a pairwise algorithm for computing sample variances". In: *COMPSTAT 5th Symposium*. Springer. 1982, pp. 30–41.

[25] Guisong Chang and Tianming Wang. "Weighted relative entropy for alignment-free sequence comparison based on Markov model". In: *Journal of Biomolecular Structure and Dynamics* 28.4 (2011), pp. 545–555.

[26] Charles Chapus et al. "Exploration of phylogenetic data using a global sequence analysis method". In: *BMC Evolutionary Biology* 5 (2005), p. 63.

[27] Nathan Clumeck, A Pozniak, and François Raffi. "European AIDS Clinical Society (EACS) guidelines for the clinical management and treatment of HIV-infected adults". In: *HIV Medicine* 9.2 (2008), pp. 65–71.

[28] Matteo Comin and Davide Verzotto. "Alignment-free phylogeny of whole genomes using underlying subwords". In: *Algorithms for Molecular Biology* 7.1 (2012), p. 34.

[29] Francis HC Crick. "On protein synthesis". In: *Symposia of the Society for Experimental Biology*. Vol. 12. 138–163. Cambridge University Press, 1958.

[30] Nello Cristianini and John Shawe-Taylor. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge University Press, 2000.

[31] Qi Dai and Tianming Wang. "Comparison study on k-word statistical measures for protein: from sequence to 'sequence space'". In: *BMC Bioinformatics* 9 (2008), p. 394.

[32] Qi Dai, Yanchun Yang, and Tianming Wang. "Markov model plus k-word distributions: A synergy that produces novel statistical measures for sequence comparison". In: *Bioinformatics* 24.20 (2008), pp. 2296–2302.

[33] Qi Dai et al. "Numerical characteristics of word frequencies and their application to dissimilarity measure for sequence comparison". In: *Journal of Theoretical Biology* 276.1 (2011), pp. 174–180.

[34] Diego Darriba et al. "jModelTest 2: more models, new heuristics and parallel computing". In: *Nature Methods* 9.8 (2012), pp. 772–772.

[35] Tulio De Oliveira et al. "An automated genotyping system for analysis of HIV-1 and other microbial sequences". In: *Bioinformatics* 21.19 (2005), pp. 3797–3800.

[36] Mo Deng et al. "A novel method of characterizing genetic sequences: Genome space with biological distance and applications". In: *PLoS One* 6.3 (2011).

[37] Patrick J Deschavanne and Miroslav Radman. "Counterselection of GATC sequences in enterobacteriophages by the components of the methyl-directed mismatch repair system". In: *Journal of Molecular Evolution* 33.2 (1991), pp. 125–132.

[38] Gilles Didier et al. "Variable length local decoding and alignment-free sequence comparison". In: *Theoretical Computer Science* 462 (2012), pp. 1–11.

[39] Mirjana Domazet-Lošo and Bernhard Haubold. "Efficient estimation of pairwise distances between genomes". In: *Bioinformatics* 25.24 (2009), pp. 3221–3227.

[40] Sanjiv K Dwivedi and Supratim Sengupta. "Classification of HIV-1 sequences using profile Hidden Markov Models". In: *PLoS One* 7.5 (2012), e36566.

[41] Betsey Dexter Dyer, Michael J Kahn, and Mark D Leblanc. "Classification and regression tree (CART) analyses of genomic signatures reveal sets of tetramers that discriminate temperature optima of archaea and bacteria". In: *Archaea* 2.3 (2008), pp. 159–167.

[42] Robert C Edgar. "MUSCLE: multiple sequence alignment with high accuracy and high throughput". In: *Nucleic Acids Research* 32.5 (2004), pp. 1792–1797.

[43] Susan H Eshleman et al. "Impact of Human Immunodeficiency Virus type 1 (HIV-1) subtype on women receiving single-dose nevirapine prophylaxis to prevent HIV-1 vertical transmission (HIV network for prevention trials 012 study)". In: *The Journal of Infectious Diseases* 184.7 (2001), pp. 914–917.

[44] Jie Feng et al. “New method for comparing DNA primary sequences based on a discrimination measure”. In: *Journal of Theoretical Biology* 266.4 (2010), pp. 703–707.

[45] Paolo Ferragina et al. “Compression-based classification of biological sequences and structures via the Universal Similarity Metric: experimental assessment”. In: *BMC Bioinformatics* 8 (2007), p. 252.

[46] Giulia Fiscon et al. “MISSEL: a method to identify a large number of small species-specific genomic subsequences and its application to viruses classification”. In: *BioData Mining* 9.38 (2016).

[47] William Fletcher and Ziheng Yang. “INDELible: a flexible simulator of biological sequence evolution”. In: *Molecular Biology and Evolution* 26.8 (2009), pp. 1879–1888.

[48] Yoav Freund and Robert E Schapire. “A decision-theoretic generalization of on-line learning and an application to boosting”. In: *Journal of Computer and System Sciences* 55.1 (1997), pp. 119–139.

[49] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. “The Elements of Statistical Learning”. In: vol. 1. New York: Springer Series in Statistics, 2001. Chap. 4.3: Linear Discriminant Analysis, pp. 106–119.

[50] Catherine V Gale et al. “Development of a novel human immunodeficiency virus type 1 subtyping tool, Subtype Analyzer (STAR): analysis of subtype distribution in London”. In: *AIDS Research and Human Retroviruses* 20.5 (2004), pp. 457–464.

[51] Yang Gao and Liaofu Luo. “Genome-based phylogeny of dsDNA viruses by a novel alignment-free method”. In: *Gene* 492.1 (2012), pp. 309–314.

[52] Mikhail S Gelfand and Eugene V Koonin. "Avoidance of palindromic words in bacterial and archaeal genomes: A close connection with restriction enzymes". In: *Nucleic Acids Research* 25.12 (1997), pp. 2430–2439.

[53] Andrew. J Gentles and Samuel Karlin. "Genome-scale compositional comparisons in Eukaryotes". In: *Genome Research* 11.4 (2001), pp. 540–546.

[54] Gene H Golub and Charles F Van Loan. *Matrix computations*. Vol. 3. JHU Press, 2012.

[55] TenoRes Study Group et al. "Global epidemiology of drug resistance after failure of WHO recommended first-line regimens for adult HIV-1 infection: a multicentre retrospective cohort study". In: *The Lancet Infectious Diseases* 16.5 (2016), pp. 565–575.

[56] Stéphane Guindon et al. "New algorithms and methods to estimate maximum-likelihood phylogenies: assessing the performance of PhyML 3.0". In: *Systematic Biology* 59.3 (2010), pp. 307–321.

[57] Frédéric Guyon, Céline Brochier-Armanet, and Alain Guénoche. "Comparison of alignment free string distances for complete genome phylogeny". In: *Advances in Data Analysis and Classification* 3.2 (2009), pp. 95–108.

[58] Isabelle Guyon et al. "Gene Selection for Cancer Classification using Support Vector Machines". In: *Machine Learning* 46.1 (2002), pp. 389–422.

[59] Trevor Hastie et al. "Multi-class AdaBoost". In: *Statistics and its Interface* 2.3 (2009), pp. 349–360.

[60] Bernhard Haubold. "Alignment-free phylogenetics and population genetics". In: *Briefings in Bioinformatics* 15.3 (2014), pp. 407–18.

[61] Bernhard Haubold and Peter Pfaffelhuber. "Alignment-free population genomics: An efficient estimator of sequence diversity". In: *G3: Genes—Genomes—Genetics* 2.8 (2012), pp. 883–889.

[62] Bernhard Haubold, Floyd A Reed, and Peter Pfaffelhuber. "Alignment-free estimation of nucleotide diversity". In: *Bioinformatics* 27.4 (2011), pp. 449–455.

[63] Bernhard Haubold et al. "Estimating mutation distances from unaligned genomes". In: *Journal of Computational Biology* 16.10 (2009), pp. 1487–1500.

[64] Paul D N Hebert et al. "Biological identifications through DNA barcodes". In: *Proceedings of the Royal Society B: Biological Sciences* 270.1512 (2003), pp. 313–321.

[65] Winston Hide, John Burke, and Daniel B Davison. "Biological evaluation of d^2, an algorithm for high-performance sequence comparison". In: *Journal of Computational Biology* 1.3 (1994), pp. 199–215.

[66] Geoffrey E Hinton. "Connectionist learning procedures". In: *Artificial Intelligence* 40.1-3 (1989), pp. 185–234.

[67] Martin S Hirsch et al. "Antiretroviral drug resistance testing in adult HIV-1 infection: 2008 recommendations of an International AIDS Society-USA panel". In: *Clinical Infectious Diseases* 47.2 (2008), pp. 266–285.

[68] Michael Höhl and Mark A Ragan. "Is multiple-sequence alignment required for accurate inference of phylogeny?" In: *Systematic Biology* 56.2 (2007), pp. 206–221.

[69] Michael Höhl, Isidore Rigoutsos, and Mark A Ragan. "Pattern-based phylogenetic distance estimation and tree reconstruction". In: *Evolutionary Bioinformatics* 2 (2006), pp. 359–375.

[70] Sebastian Horwege et al. "Spaced words and kmacs: Fast alignment-free sequence comparison based on inexact word matches". In: *Nucleic Acids Research* 42.W1 (2014), W7–W11.

[71] Diana D Huang, Thomas A Giesler, and James W Bremer. "Sequence characterization of the protease and partial reverse transcriptase proteins of the NED panel, an international HIV type 1 subtype reference and standards panel". In: *AIDS Research and Human Retroviruses* 19.4 (2003), pp. 321–328.

[72] Hsin-Hsiung Huang et al. "Global comparison of multiple-segmented viruses in 12-dimensional genome space". In: *Molecular Phylogenetics and Evolution* 81.Supplement C (2014), pp. 29–36.

[73] Yujuan Huang, Lianping Yang, and Tianming Wang. "Phylogenetic analysis of DNA sequences based on the generalized pseudo-amino acid composition". In: *Journal of Theoretical Biology* 269.1 (2011), pp. 217–223.

[74] Arihant Kumar Jain and B. Chandrasekaran. "39 dimensionality and sample size considerations in pattern recognition practice". In: *Classification Pattern Recognition and Reduction of Dimensionality*. Vol. 2. Handbook of Statistics. Elsevier, 1982, pp. 835–855.

[75] Ramamurthy Jayalakshmi et al. "Alignment-free sequence comparison using N-dimensional similarity space". In: *Current Computer-Aided Drug Design* 6.4 (2010), pp. 290–296.

[76] H Joel Jeffrey. "Chaos game representation of gene structure". In: *Nucleic Acids Research* 18.8 (1990), pp. 2163–2170.

[77] Robert W Jernigan and Robert H Baran. "Pervasive properties of the genomic signature". In: *BMC Genomics* 3.1 (2002), p. 23.

[78] Jeffrey B Joy et al. "Origin and evolution of Human Immunodeficiency Viruses". In: *Global Virology I-Identifying and Investigating Viral Diseases*. Springer, 2015, pp. 587–611.

[79] Rallis Karamichalis and Lila Kari. "MoDMaps3D: an interactive webtool for the quantification and 3D visualization of interrelationships in a dataset of DNA sequences". In: *Bioinformatics* 33.19 (2017), pp. 3091–3093.

[80] Rallis Karamichalis et al. "Additive methods for genomic signatures". In: *BMC Bioinformatics* 17.1 (2016), p. 313.

[81] Rallis Karamichalis et al. "An investigation into inter- and intragenomic variations of graphic genomic signatures". In: *BMC Bioinformatics* 16.1 (2015), p. 246.

[82] Lila Kari et al. "Mapping the space of genomic signatures". In: *PLoS One* 10.5 (2015).

[83] Samuel Karlin. "Global dinucleotide signatures and analysis of genomic heterogeneity". In: *Current Opinion in Microbiology* 1.5 (1998), pp. 598–610.

[84] Samuel Karlin and Chris Burge. "Dinucleotide relative abundance extremes: a genomic signature". In: *Trends in Genetics* 11.7 (1995), pp. 283–290.

[85] Samuel Karlin, Chris Burge, and Allan M Campbell. "Statistical analyses of counts and distributions of restriction sites in DNA sequences". In: *Nucleic Acids Research* 20.6 (1992), pp. 1363–1370.

[86] Samuel Karlin, Allan M Campbell, and J Mrázek. "Comparative DNA analysis across diverse genomes". In: *Annual Review of Genetics* 32 (1998), pp. 185–225.

[87] Samuel Karlin and István Ladunga. "Comparisons of eukaryotic genomic sequences". In: *Proceedings of the National Academy of Sciences of the United States of America* 91.26 (1994), pp. 12832–12836.

[88] Samuel Karlin, István Ladunga, and B Edwin Blaisdell. "Heterogeneity of genomes: measures and values". In: *Proceedings of the National Academy of Sciences of the United States of America* 91.26 (1994), pp. 12837–12841.

[89] Samuel Karlin, Jan Mrázek, and Allan M Campbell. "Compositional biases of bacterial genomes and evolutionary implications". In: *Journal of Bacteriology* 179.12 (1997), pp. 3899–3913.

[90] Patrick J. Keeling et al. "The Marine Microbial Eukaryote Transcriptome Sequencing Project (MMETSP): Illuminating the Functional Diversity of Eukaryotic Life in the Oceans through Transcriptome Sequencing". In: *PLOS Biology* 12.6 (2014), pp. 1–6.

[91] Diederik Kingma and Jimmy Ba. "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

[92] Pandurang Kolekar, Mohan Kale, and Urmila Kulkarni-Kale. "Alignment-free distance measure based on return time distribution for sequence analysis: Applications to clustering, molecular phylogeny and subtyp-

ing". In: *Molecular Phylogenetics and Evolution* 65.2 (2012), pp. 510–522.

[93] Natalio Krasnogor and David A Pelta. "Measuring the similarity of protein structures by means of the Universal Similarity Metric". In: *Bioinformatics* 20.7 (2004), pp. 1015–1021.

[94] Eugene F Krause. *Taxicab geometry: An adventure in non-Euclidean geometry*. Mineola, New York: Courier Dover Publications, 2012.

[95] Carla Kuiken et al. *HIV sequence compendium 2010*. Tech. rep. Los Alamos National Laboratory (LANL), Los Alamos, NM (United States), 2010.

[96] Anders Larsson. "AliView: a fast and lightweight alignment viewer and editor for large datasets". In: *Bioinformatics* 30.22 (2014), pp. 3276–3278.

[97] Chris Andre Leimeister and Burkhard Morgenstern. "Kmacs: The k-mismatch average common substring approach to alignment-free sequence comparison". In: *Bioinformatics* 30.14 (2014), pp. 2000–2008.

[98] Chris Andre Leimeister et al. "Fast alignment-free sequence comparison using spaced-word frequencies". In: *Bioinformatics* 30.14 (2014), pp. 1991–1999.

[99] Thomas Leitner et al. "HIV1 subtype and circulating recombinant form (CRF) reference sequences, 2005". In: *HIV sequence compendium 2005*. Los Alamos National Laboratory, pp. 41–48.

[100] Bin Li, Yi Bing Li, and Hong Bo He. "LZ complexity distance of DNA sequences and its application in phylogenetic tree reconstruction". In: *Genomics, Proteomics & Bioinformatics* 3.4 (2005), pp. 206–212.

[101] Yongkun Li et al. “Virus classification in 60-dimensional protein space”. In: *Molecular Phylogenetics and Evolution* 99.Supplement C (2016), pp. 53–62.

[102] DJ Lipman and WR Pearson. “Rapid and sensitive protein similarity searches”. In: *Science* 227.4693 (1985), pp. 1435–1441.

[103] Jingjun Liu and Dachao Li. “Conditional LZ complexity of DNA sequences analysis and its application in phylogenetic tree reconstruction”. In: *Proceedings of the International Conference on BioMedical Engineering and Informatics*. 2008, pp. 111–116.

[104] Zhi Hua Liu and Xiao Sun. “Coronavirus phylogeny based on base-base correlation”. In: *International Journal of Bioinformatics Research and Applications* 4.2 (2008), pp. 211–220.

[105] Zhi Hua Liu et al. “Base-Base Correlation a novel sequence feature and its applications”. In: *Proceedings of the 1st International Conference on Bioinformatics and Biomedical Engineering*. 2007, pp. 370–373.

[106] Zhihua Liu, Jihong Meng, and Xiao Sun. “A novel feature-based method for whole genome phylogenetic analysis without alignment: application to HEV genotyping and subtyping”. In: *Biochemical and Biophysical Research Communications* 368.2 (2008), pp. 223–230.

[107] Marie T. Lott et al. “mtDNA Variation and Analysis Using Mitomap and Mitomaster”. In: *Current Protocols in Bioinformatics* 44.1 (2013), pp. 1.23.1–1.23.26.

[108] Christian Martin et al. “Hyperbolic SOM-based clustering of DNA fragment features for taxonomic visualization and classification”. In: *Bioinformatics* 24.14 (2008), pp. 1568–1574.

[109] Naim Matasci et al. “Data access for the 1,000 Plants (1KP) project”. In: *GigaScience* 3.1 (2014), pp. 1–10.

[110] David W Mount. *Bioinformatics: Sequence and Genome Analysis*. 2nd ed. Cold Spring Harbor Laboratory Press, New York, 2004.

[111] Yuka Nadai et al. “HIV-1 epidemic in the Caribbean is dominated by subtype B”. In: *PLoS One* 4.3 (2009), e4814.

[112] Vrinda V. Nair and Achuthsankar S. Nair. “Combined classifier for unknown genome classification using Chaos Game Representation features”. In: *Proceedings of the International Symposium on Biocomputing: ISB ’10*. New York, NY, USA: ACM, 2010, 35:1–35:8.

[113] Vrinda V. Nair et al. “Hurst CGR (HCGR) – A novel feature extraction method from Chaos Game Representation of genomes”. In: *Proceedings of the First International Conference on Advances in Computing and Communications: ACC 2011*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 302–309.

[114] Vrinda V Nair et al. “ANN based classification of unknown genome fragments using Chaos Game Representation”. In: *Second International Conference on Machine Learning and Computing (ICMLC 2010)*. IEEE. 2010, pp. 81–85.

[115] Vrinda V Nair et al. “Texture features from Chaos Game Representation images of genomes”. In: *International Journal of Image Processing* 7.2 (2013), pp. 183–190.

[116] Ozkan Ufuk Nalbantoglu and Khalid Sayood. “Computational genomic signatures”. In: *Synthesis Lectures on Biomedical Engineering* 6.2 (2011), pp. 1–129.

[117] Ngoc Giang Nguyen et al. “DNA Sequence Classification by Convolutional Neural Network”. In: *Journal of Biomedical Science and Engineering* 9.5 (2016), pp. 280–286.

[118] Iulia Niculescu et al. “Recent HIV-1 outbreak among intravenous drug users in Romania: evidence for cocirculation of CRF14_BG and subtype F1 strains”. In: *AIDS Research and Human Retroviruses* 31.5 (2015), pp. 488–495.

[119] Hasan H Otu and Khalid Sayood. “A new sequence distance measure for phylogenetic tree construction”. In: *Bioinformatics* 19.16 (2003), pp. 2122–2130.

[120] Aridaman Pandit, Anil Kumar Dasanna, and Somdatta Sinha. “Multifractal analysis of HIV-1 genomes”. In: *Molecular Phylogenetics and Evolution* 62.2 (2012), pp. 756–763.

[121] Aridaman Pandit, Jyothirmayi Vadlamudi, and Somdatta Sinha. “Analysis of dinucleotide signatures in HIV-1 subtype B genomes”. In: *Journal of Genetics* 92.3 (2013), pp. 403–412.

[122] Simona Paraschiv et al. “Epidemic dispersion of HIV and HCV in a population of co-infected Romanian injecting drug users”. In: *PLoS One* 12.10 (2017), e0185866.

[123] Kaustubh R Patil and Alice C McHardy. “Alignment-free genome tree inference by learning group-specific distance metrics”. In: *Genome Biology and Evolution* 5.8 (2013), pp. 1470–1484.

[124] Nagamma Patil, Durga Toshniwal, and Kumkum Garg. “Species Identification Based on Approximate Matching”. In: *Proceedings of the Fourth Annual ACM Bangalore Conference*. ACM, 2011, 30:1–30:4.

[125] Fabian Pedregosa et al. “Scikit-learn: Machine Learning in Python”. In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[126] Tuan D Pham. “Spectral distortion measures for biological sequence comparisons and database searching”. In: *Pattern Recognition* 40.2 (2007), pp. 516–529.

[127] Tuan D Pham and Johannes Zuegg. “A probabilistic measure for alignment-free sequence comparison”. In: *Bioinformatics* 20.18 (2004), pp. 3455–3461.

[128] Dau Phan et al. “Combined Use of k-Mer Numerical Features and Position-Specific Categorical Features in Fixed-Length DNA Sequence Classification”. In: *Journal of Biomedical Science and Engineering* 10.8 (2017), pp. 390–401.

[129] Andrea-Clemencia Pineda-Peña et al. “Automated subtyping of HIV-1 genetic sequences for clinical and surveillance purposes: performance evaluation of the new REGA version 3 and seven other tools”. In: *Infection, Genetics and Evolution* 19 (2013), pp. 337–348.

[130] Sergei L Kosakovsky Pond et al. “An evolutionary model-based algorithm for accurate phylogenetic breakpoint mapping and subtype prediction in HIV-1”. In: *PLoS Computational Biology* 5.11 (2009), e1000581.

[131] Art F. Y. Poon. “Phylodynamic inference with kernel ABC and its application to HIV epidemiology”. In: *Molecular Biology and Evolution* 32.9 (2015), pp. 2483–2495.

[132] David T Pride et al. “Evolutionary implications of microbial genome tetranucleotide frequency biases”. In: *Genome Research* 13.2 (2003), pp. 145–156.

[133] Kim D. Pruitt et al. “RefSeq: an update on mammalian reference sequences”. In: *Nucleic Acids Research* 42.D1 (2014), pp. D756–D763.

[134] Andrew Rambaut. *FigTree*. 2016. URL: http://tree.bio.ed.ac.uk/software/figtree/.

[135] Sarunas J. Raudys and Arihant Kumar Jain. “Small sample size effects in statistical pattern recognition: Recommendations for practitioners”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13.3 (Mar. 1991), pp. 252–264.

[136] Payam Refaeilzadeh, Lei Tang, and Huan Liu. “Cross-Validation”. In: *Encyclopedia of Database Systems*. Ed. by Ling Liu and M. Tamer Özsu. Boston, MA: Springer US, 2009, pp. 532–538.

[137] Mohamed Amine Remita et al. “A machine learning approach for viral genome classification”. In: *BMC Bioinformatics* 18.208 (2017).

[138] Soo-Yon Rhee et al. “Mutational correlates of virological failure in individuals receiving a WHO-recommended tenofovir-containing first-line regimen: An international collaboration”. In: *EBioMedicine* 18 (2017), pp. 225–235.

[139] David L. Robertson et al. “HIV-1 nomenclature proposal”. In: *Science* 288.5463 (2000), pp. 55–55.

[140] Mikhail Rozanov et al. “A web-based genotyping resource for viral sequences”. In: *Nucleic Acids Research* 32.suppl_2 (2004), W654–W659.

[141] Mika O Salminen et al. “Identification of breakpoints in intergenotypic recombinants of HIV type 1 by bootscanning”. In: *AIDS Research and Human Retroviruses* 11.11 (1995), pp. 1423–1425.

[142] Rickard Sandberg et al. “Capturing whole-genome characteristics in short sequences using a naıve Bayesian classifier”. In: *Genome Research* 11.8 (2001), pp. 1404–1409.

[143] Anne-Kathrin Schultz et al. “A jumping profile Hidden Markov Model and applications to recombination sites in HIV and HCV genomes”. In: *BMC Bioinformatics* 7.1 (2006), p. 265.

[144] Isabel Schwende and Tuan D Pham. “Pattern recognition and probabilistic measures in alignment-free sequence analysis”. In: *Briefings in Bioinformatics* 15.3 (2014), pp. 354–68.

[145] Andrew M Shedlock et al. “Phylogenomics of nonavian reptiles and the structure of the ancestral amniote genome”. In: *Proceedings of the National Academy of Sciences of the United States of America* 104.8 (2007), pp. 2767–2772.

[146] Peter Simmonds et al. “Consensus proposals for a unified system of nomenclature of hepatitis C virus genotypes”. In: *Hepatology* 42.4 (2005), pp. 962–73.

[147] Gregory E Sims and Sung-Hou Kim. “Whole-genome phylogeny of Escherichia coli/Shigella group by feature frequency profiles (FFPs)”. In: *Proceedings of the National Academy of Sciences of the United States of America* 108.20 (2011), pp. 8329–8334.

[148] Gregory E Sims et al. “Alignment-free genome comparison with feature frequency profiles (FFP) and optimal resolutions”. In: *Proceedings of the National Academy of Sciences of the United States of America* 106.8 (2009), pp. 2677–2682.

[149] Kai Song et al. “New developments of alignment-free sequence comparison: Measures, statistics and next-generation sequencing”. In: *Briefings in Bioinformatics* 15.3 (2014), pp. 343–353.

[150] Deogratius Ssemwanga et al. “Low drug resistance levels among drug-naive individuals with recent HIV type 1 infection in a rural clinical cohort in southwestern Uganda”. In: *AIDS Research and Human Retroviruses* 28.12 (2012), pp. 1784–1787.

[151] Daniel Struck et al. “COMET: adaptive context-based modeling for ultrafast HIV-1 subtype identification”. In: *Nucleic Acids Research* 42.18 (2014), e144–e144.

[152] Chonlaphat Sukasem et al. “Surveillance of genotypic resistance mutations in chronic HIV-1 treated individuals after completion of the National Access to Antiretroviral Program in Thailand”. In: *Infection* 35.2 (2007), pp. 81–88.

[153] Haruo Suzuki et al. “Using Mahalanobis distance to compare genomic signatures between bacterial plasmids and chromosomes”. In: *Nucleic Acids Research* 36.22 (2008).

[154] Watcharaporn Tanchotsrinon, Chidchanok Lursinsap, and Yong Poovorawan. “A high performance prediction of HPV genotypes by Chaos game representation and singular value decomposition”. In: *BMC Bioinformatics* 16.1 (2015).

[155] Watcharaporn Tanchotsrinon, Chidchanok Lursinsap, and Yong Poovorawan. “An efficient prediction of HPV genotypes from partial coding sequences by Chaos Game Representation and fuzzy k-nearest neighbor technique”. In: *Current Bioinformatics* 12.5 (2017), pp. 431–440.

[156] Barbara S Taylor et al. “The challenge of HIV-1 subtype diversity”. In: *New England Journal of Medicine* 358.15 (2008), pp. 1590–1602.

[157] Denis M Tebit and Eric J Arts. “Tracking a century of global expansion and evolution of HIV to drive understanding and to combat disease”. In: *The Lancet Infectious Diseases* 11.1 (2011), pp. 45–56.

[158] Hanno Teeling et al. “Application of tetranucleotide frequencies for the assignment of genomic fragments”. In: *Environmental Microbiology* 6.9 (2004), pp. 938–947.

[159] Julie D. Thompson, Desmond G. Higgins, and Toby J. Gibson. “CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice”. In: *Nucleic Acids Research* 22.22 (1994), pp. 4673–4680.

[160] Robert Tibshirani et al. “Diagnosis of multiple cancer types by shrunken centroids of gene expression”. In: *Proceedings of the National Academy of Sciences of the United States of America* 99.10 (2002), pp. 6567–6572.

[161] Roberto Todeschini et al. “Characterization of DNA primary sequences by a new similarity/diversity measure based on the partial ordering”. In: *Journal of Chemical Information and Modeling* 46.5 (2006), pp. 1905–1911.

[162] Igor Ulitsky et al. “The average common substring approach to phylogenomic reconstruction”. In: *Journal of Computational Biology* 13.2 (2006), pp. 336–350.

[163] Susana Vinga and Jonas S Almeida. “Alignment-free sequence comparison – A review”. In: *Bioinformatics* 19.4 (2003), pp. 513–523.

[164] Arnoud H M van Vliet and Johannes G Kusters. “Use of alignment-free phylogenetics for rapid genome sequence-based typing of Helicobacter pylori virulence markers and antibiotic susceptibility”. In: *Journal of clinical microbiology* 53.9 (2015), pp. 2877–2888.

[165] Jing-Doo Wang. “Comparing virus classification using genomic materials according to different taxonomic levels”. In: *Journal of Bioinformatics and Computational Biology* 11.06 (2013), p. 1343003.

[166] Wenwen Wang and Tianming Wang. “Conditional LZ complexity and its application in mtDNA sequence analysis”. In: *MATCH Communications in Mathematical and in Computer Chemistry* 66.1 (2011), pp. 425–443.

[167] Emanuel Weitschek, Fabio Cunial, and Giovanni Felici. “LAF: Logic Alignment Free and its application to bacterial genomes classification”. In: *BioData Mining* 8.39 (2015).

[168] Elizabeth Wolf et al. “Phylogenetic evidence of HIV-1 transmission between adult and adolescent men who have sex with men”. In: *AIDS Research and Human Retroviruses* 33.4 (2017), pp. 318–322.

[169] Michael Worobey et al. “Direct evidence of extensive diversity of HIV-1 in Kinshasa by 1960”. In: *Nature* 455.7213 (2008), pp. 661–4.

[170] Guohong Albert Wu et al. “Whole-proteome phylogeny of large dsDNA virus families by an alignment-free method”. In: *Proceedings of the National Academy of Sciences of the United States of America* 106.31 (2009), pp. 12826–12831.

[171] Tiee Jian Wu, John P Burke, and Daniel B Davison. “A measure of DNA sequence dissimilarity based on Mahalanobis distance between frequencies of words”. In: *Biometrics* 53.4 (1997), pp. 1431–1439.

[172] Tiee Jian Wu, Ying Hsueh Hsieh, and Lung A Li. "Statistical measures of DNA sequence dissimilarity under Markov chain models of base composition". In: *Biometrics* 57.2 (2001), pp. 441–448.

[173] Tiee Jian Wu, Ying Hsueh Huang, and Lung A Li. "Optimal word sizes for dissimilarity measures and estimation of the degree of dissimilarity between DNA sequences". In: *Bioinformatics* 21.22 (2005), pp. 4125–4132.

[174] Lianping Yang et al. "Use of the Burrows-Wheeler similarity distribution to the comparison of the proteins". In: *Amino Acids* 39.3 (2010), pp. 887–898.

[175] Xiwu Yang and Tianming Wang. "A novel statistical measure for sequence comparison on the basis of k-word counts". In: *Journal of Theoretical Biology* 318 (2013), pp. 91–100.

[176] Huiguang Yi and Li Jin. "Co-phylog: An assembly-free phylogenomic approach for closely related organisms". In: *Nucleic Acids Research* 41.7 (2013), p. 75.

[177] Chenglong Yu et al. "Real time classification of viruses in 12 dimensions". In: *PLoS One* 8.5 (2013).

[178] Zu-Guo Yu, Vo Anh, and Ka-Sing Lau. "Measure representation and multifractal analysis of complete genomes". In: *Physical Review E: Statistical, Nonlinear, and Soft Matter Physics* 64.3 (2001), p. 031903.

[179] Zu-Guo Yu, Vo Anh, and Ka-Sing Lau. "Multifractal and correlation analyses of protein sequences from complete genomes". In: *Physical Review E: Statistical, Nonlinear, and Soft Matter Physics* 68.2 (2003), p. 021913.

[180] Tong Zhang. "Solving large scale linear prediction problems using stochastic gradient descent algorithms". In: *ICML 2004: Proceedings Of The Twenty-First International Conference On Machine Learning*. Omnipress, 2004, pp. 919–926.

[181] Yusen Zhang and Wei Chen. "A measure of DNA sequence dissimilarity based on free energy of nearest-neighbor interaction". In: *Journal of Biomolecular Structure and Dynamics* 28.4 (2011), pp. 557–565.

[182] Andrzej Zielezinski et al. "Alignment-free sequence comparison: benefits, applications, and tools". In: *Genome Biology* 18.186 (2017).

[183] Gert U van Zyl et al. "Moderate levels of preantiretroviral therapy drug resistance in a generalized epidemic: time for better first-line ART?" In: *AIDS* 31.17 (2017), pp. 2387–2391.

Appendix A

Instructions for reproduction of the experiments from Chapter 3 using Kameris

First, visit https://github.com/stephensolis/kameris and follow the instructions for installing KAMERIS. It is highly recommended you also work through the demo instructions on the same page, to understand how the software works.

10-fold cross-validation on the full set of full-length HIV-1 genomes

Run `kameris run-job https://raw.githubusercontent.com/stephensolis/kameris-experiments/master/experiments/hiv1/lanl-whole.yml https://raw.githubusercontent.com/stephensolis/kameris/master/demo/settings.yml`

10-fold cross-validation on the full set of HIV-1 *pol* genes

Run `kameris run-job https://raw.githubusercontent.com/stephensolis/kameris-experiments/master/experiments/hiv1/lanl-pol.yml https://raw.githubusercontent.com/stephensolis/kameris/master/demo/settings.yml`

Classification of the HIV-1 benchmark dataset

1. Run `kameris run-job https://raw.githubusercontent.com/stephensolis/kameris-experiments/master/experiments/hiv1/lanl-reference-model.yml https://raw.githubusercontent.com/stephensolis/kameris/master/demo/settings.yml` to train the model.
2. Download `https://drive.google.com/uc?export=download&id=0B7OX388vZjTva1NNYXh5WEM2Z28` and extract the `mixed-polfragments` folder.
3. Run `kameris classify output/lanl-reference-model/subtype-k=6/model_linear-svm.mm-model "path to the mixed-polfragments folder"`
4. Compare the output stored in `results.json` with the ground-truth subtypes from https://raw.githubusercontent.com/stephensolis/kameris-experiments/master/metadata/hiv1-mixed-polfragments.json.

10-fold cross-validation on the synthetic-vs-natural HIV-1 *pol* genes

Run `kameris run-job https://raw.githubusercontent.com/stephensolis/kameris-experiments/master/experiments/hiv1/real-vs-synthetic.yml`

`https://raw.githubusercontent.com/stephensolis/kameris/master/demo/settings.yml`

Classification of randomly-generated sequence

1. Follow steps 1-2 from Classification of the HIV-1 benchmark dataset.
2. Generate a random sequence, for example using http://www.faculty.ucr.edu/~mmaduro/random.htm, save it to a file, and put the file in a new folder by itself.
3. Run `kameris classify output/lanl-reference-model/subtype-k=6/model_linear-svm.mm-model "path to the folder you created"`

10-fold cross-validation on the set of whole dengue virus genomes

Run `kameris run-job https://raw.githubusercontent.com/stephensolis/kameris-experiments/master/experiments/dengue/ncbi-whole.yml https://raw.githubusercontent.com/stephensolis/kameris/master/demo/settings.yml`

10-fold cross-validation on the set of whole hepatitis B genomes

Run `kameris run-job https://raw.githubusercontent.com/stephensolis/kameris-experiments/master/experiments/hepatitis/hbv-whole.yml https://raw.githubusercontent.com/stephensolis/kameris/master/demo/settings.yml`

10-fold cross-validation on the set of whole hepatitis C genomes

Run `kameris run-job https://raw.githubusercontent.com/stephensolis/kameris-experiments/master/experiments/hepatitis/hcv-whole.yml https://raw.githubusercontent.com/stephensolis/kameris/master/demo/settings.yml`

10-fold cross-validation on the set of whole influenza A genomes

Run `kameris run-job https://raw.githubusercontent.com/stephensolis/kameris-experiments/master/experiments/flu/ncbi-whole.yml https://raw.githubusercontent.com/stephensolis/kameris/master/demo/settings.yml`

Appendix B

Lists of subtypes of viral species from datasets used

Whole HIV-1 genomes

Subtype	**Recombinant?**
01B	Yes
01BC	Yes
01_AE	Yes
02A1	Yes
02_AG	Yes
07_BC	Yes
08_BC	Yes
11_cpx	Yes
35_AD	Yes
A1	No
A1C	Yes
A1CD	Yes
A1D	Yes
A6	No
B	No
BC	Yes
BF	Yes
BF1	Yes
C	No
CD	Yes
D	No
F1	No
G	No
O	No
U	No

Full set of HIV-1 *pol* genes

Subtype	**Recombinant?**
0107	Yes
01B	Yes
01BC	Yes
01_AE	Yes
02A1	Yes
02_AG	Yes
07_BC	Yes
08_BC	Yes
11_cpx	Yes
35_AD	Yes
A1	No
A1C	Yes
A1CD	Yes
A1D	Yes
A6	No
B	No
BC	Yes
BF	Yes
BF1	Yes
C	No
CD	Yes
D	No
F1	No
G	No
O	No
U	No

HIV-1 *pol* genes from the 2010 LANL Web alignment

Subtype	**Recombinant?**
01B	Yes
01_AE	Yes
02_AG	Yes
A1	No
A1C	Yes
A1D	Yes
B	No
BC	Yes
BF	Yes
BF1	Yes
C	No
D	No
F1	No
G	No
O	No

Whole dengue virus genomes

Subtype
1
2
3
4

Whole hepatitis B genomes

Subtype	Recombinant?
A	No
B	No
C	No
D	No
E	No
F	No
G	No
H	No
RF-BC	Yes
RF-CB	Yes
RF-DC	Yes
RF-DE	Yes

Whole hepatitis C genomes

Subtype
1a
1b
2a
2b
3a
6a

Whole influenza A genomes

Subtype	Subtype
H1N1	H6N6
H1N2	H6N8
H1N3	H7N1
H1N6	H7N2
H1N9	H7N3
H2N1	H7N4
H2N2	H7N6
H2N3	H7N7
H2N7	H7N9
H2N9	H8N4
H3N1	H9N2
H3N2	H10N1
H3N3	H10N3
H3N6	H10N4
H3N8	H10N5
H4N2	H10N6
H4N6	H10N7
H4N8	H10N8
H4N9	H11N1
H5N1	H11N2
H5N2	H11N3
H5N3	H11N9
H5N5	H12N5
H5N6	H13N2
H5N8	H13N6
H6N1	H13N8
H6N2	H16N3

Curriculum Vitae

Name:	Stephen Solis-Reyes
Post-Secondary Education and Degrees:	B.Sc. in Computer Science University of Western Ontario London, ON, Canada, 2012 – 2016
Honours and Awards:	NSERC Undergraduate Student Research Award (USRA) 2014 – 2016 IBM Master the Mainframe 2016 World Championship, 2nd place winner
Related Work Experience:	Director of Technology, Infrastructure, and Security Triage Technologies, Inc. 2018 – present Graduate Teaching Assistant The University of Western Ontario 2016 – 2017

Publications:

R. Karamichalis, L. Kari, S. Konstantinidis, S. Kopecki, S. Solis-Reyes. Additive methods for genomic signatures. BMC Bioinformatics, 17:313, 2016.