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

Stephen Solis-Reyes [1] Mariano Avino [2], Art Poon [2,Y], and Lila Kari [3,Y,*]

1 Department of Computer Science, University of Western Ontario, London, ON, N6A 5B7, Canada
2 Department of Pathology and Laboratory Medicine, University of Western Ontario, London, ON, N6A 5C1, Canada
3 School of Computer Science, University of Waterloo, Waterloo, ON, N2L 3G1, Canada

[Y] Senior authors
[*] Corresponding author, lila@uwaterloo.ca

Abstract

For many disease-causing virus species, global diversity is clustered into a taxonomy of subtypes with clinical significance. In particular, the classification of infections among the subtypes of human immunodeficiency virus type 1 (HIV-1) is a routine component of clinical management, and there are now many classification algorithms available for this purpose. Although several of these algorithms are similar in accuracy and speed, the majority are proprietary and require laboratories to transmit HIV-1 sequence data over the network to remote servers. This potentially exposes sensitive patient data to unauthorized access, and makes it impossible to determine how classifications are made and to maintain the data provenance of clinical bioinformatic workflows. We propose an open-source supervised and alignment-free subtyping method (Kameris) that operates on k-mer frequencies in HIV-1 sequences. We performed a detailed study of the accuracy and performance of subtype classification in comparison to four state-of-the-art programs. Based on our testing data set of manually curated real-world HIV-1 sequences ($n = 2,784$), Kameris obtained an overall accuracy of 97%, which matches or exceeds all other tested software, with a processing rate of over 1,500 sequences per second. Furthermore, our fully standalone general-purpose software provides key advantages in terms of data security and privacy, transparency and reproducibility. Finally, we show that our method is readily adaptable to subtype classification of other viruses including dengue, influenza A, and hepatitis B and C virus.

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 [1], but there is no consistent threshold among virus species. Many virus subtypes are clinically significant because of their associations with variation in pathogenesis, rates of disease progression, and susceptibility to drug treatments and vaccines [2]. For example, the HIV-1 subtypes originated early in the history of the global pandemic [3] and have diverged by about 15% of the nucleotide genome sequence [4]. Rates of disease progression vary significantly among HIV-1 subtypes and classifying newly diagnosed infections by their genetic similarity to curated reference subtypes [5] is a recommended component for the clinical management of HIV-1 infection [6, 7]. Consequently, a number of algorithms have been developed for the automated determination of HIV-1 subtypes from genetic sequence data [8–10].

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 frequency 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.

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 [11], which enables the algorithm to compare homologous sequence features [12–14]. For example, the NCBI genotyping tool [15] computes BLAST similarity scores against the reference set for sliding windows along the query sequence. Other methods such as REGA [9] and SCUEAL [10] reconstruct maximum likelihood phylogenies from the aligned sequences: REGA (version 3.0) reconstructs trees from sliding windows of 400bp from the sequence alignment and quantifies the confidence in placement of the query sequence within subtypes by bootstrap sampling (bootscanning) [16]. 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 [17], or sequence composition (*e.g.* COMET [8] and [18]). 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 [19]); 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 [18] and multi-segmented [20] whole viral genomes, as well as viral proteomes [21]; based on neural networks using digital signal processing techniques to yield "genomic cepstral coefficient" features, applied to distinguishing four different pathogenic viruses [22]; 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 [23].

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 nDNA sequences [24] and some mammalian alpha and beta-globin genes [25]. Other authors [26–30] have observed that the excess and scarcity of specific k-mers, across a variety of different DNA sequence types (including viral DNA in [26]), 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 vectors are paired together with a distance function in order to measure the quantitative similarity between any pair of sequences. Studies measuring quantitative similarity between DNA sequences from different sources have been performed, for instance using the Manhattan distance [31, 32], the weighted or standardized Euclidean distance [33, 34], and the Jensen-Shannon distance [35, 36]. Applications of these distances and others have been compared and benchmarked in [37–40], and detailed reviews of the literature can be found in [41–44].

In the context of viral phylogenetics, k-mer frequency vectors paired with a distance metric have been used to construct pairwise distance matrices and derive phylogenetic trees, *e.g.*, dsDNA eukaryotic viruses [45], or fragments from Flaviviridae genomes [46]. Other studies have investigated the multifractal properties of k-mer frequency patterns in HIV-1 genomes [47], and the changes in dinucleotide frequencies in the HIV genome across different years [48]. We used k-mer frequency 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 [49], to predict HPV genotypes [50, 51], to classify whole bacterial genomes to their corresponding taxonomic groups at different levels [52], and to classify whole eukaryotic mitochondrial genomes [53–56].

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 $k = 1 \dots 10$ 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 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.

Methods

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 frequencies as feature vectors (numerical representations). We trained each of 15 classifiers (Table 2) on a set $S = \{s_1, s_2, \dots s_n\}$ of nucleotide sequences partitioned into groups $g_1, g_2, \dots, g_p$. 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 frequencies 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 $F_k(s)$.

Next, we processed the feature vectors for more efficient use by classifiers. We rescaled the k-mer frequencies in $F_k(s)$ to have a standard deviation of 1, which satisfies some statistical assumptions invoked by several classification methods. In addition, we performed dimensionality reduction using truncated singular value decomposition [57] 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 [58] with Euclidean metric (`10-nearest-neighbors`); nearest centroid, to class mean (`nearest-centroid-mean`) and to class median (`nearest-centroid-median`) [59]; logistic regression with L2 regularization and one-vs-rest as the multiclass generalization (`logistic-regression`) [60]; SVM with the linear (`linear-svm`), quadratic (`quadratic-svm`), and cubic (`cubic-svm`) kernel functions [61]; SVM with stochastic gradient descent learning and linear kernel function (`sgd`) [62]; decision tree with Gini impurity metric (`decision-tree`) [63]; random forest using decision trees with Gini impurity metric as sub-estimators (`random-forest`) [64]; AdaBoost with decision trees as the weak learners and the SAMME.R real boosting algorithm (`adaboost`) [65, 66]; Gaussian naïve Bayes (`gaussian-naive-bayes`) [67]; linear (`lda`) and quadratic (`qda`) discriminant analysis [68]; 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`) [69, 70]. We used the implementations of these classifiers in the Python library `scikit-learn` [71] 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 [72]. 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.

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 [40, 73, 74], 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 [75], defined between two vectors $A = (a_1, \dots a_n)$ and $B = (b_1, \dots b_n)$ 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) [76]. MDS takes as input a pairwise distance matrix and produces as output a 2D or 3D plot, called a MoDMap [77], 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.

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. Detailed documentation, 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.

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 easily retrieved from our GitHub repository at `https://github.com/stephensolis/kameris-experiments`.

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 [78, 79]. Since we vary both k and the classifier algorithms in this study, this makes it especially challenging to 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 26 subtypes and circulating recombinant forms (CRFs). 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 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 [80]. 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` [81].

Table 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) [82]	Haiti	B	66	1024.0	1024	1025
Niculescu (2015) [83]	Romania	F	97	1301.2	1257	1302
Paraschiv (2017) [84]	Romania	F	86	1295.9	1164	1299
Rhee (2017) [85]	Thailand	CRF01_AE	282	703.8	633	756
Sukasem (2007) [86]	Thailand	CRF01_AE	221	286.4	270	288
Eshleman (2001) [87]	Uganda	A/D	102	1261.2	1260	1302
Ssemwanga (2012) [88]	Uganda	A/D	72	1025.0	1025	1025
Wolf (2017) [89]	USA	B	1653	1020.8	868	1080
TenoRes Study Group (2016) [90]	South Africa	C	102	1001.4	921	1209
van Zyl (2017) [91]	South Africa	C	59	1056.7	1002	1070
Huang (2003) [92]	Reference panel	n/a	44	1189.9	1187	1190
Overall			2784	960.4	270	1302

We loaded the resulting FASTA file in the eleven datasets from Table 1. We then aligned the datasets with MUSCLE v3.8.425 [93], implemented in AliView 1.19-beta-3 [94], 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 [95]. 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 [89]. Phylogenetic trees were reconstructed by maximum likelihood using PHYML v20160207 [96] with a related bootstrap support analysis. The resulting trees were visualized and their relative sequences were manually annotated in FigTree v1.4.3 [97].

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 'Predefined 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 16 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 [98], 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 [99]. 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 sequences 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, 6 subtypes of hepatitis B, 12 subtypes of hepatitis C, and 56 subtypes of influenza type A.

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 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 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.

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

Fig 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.

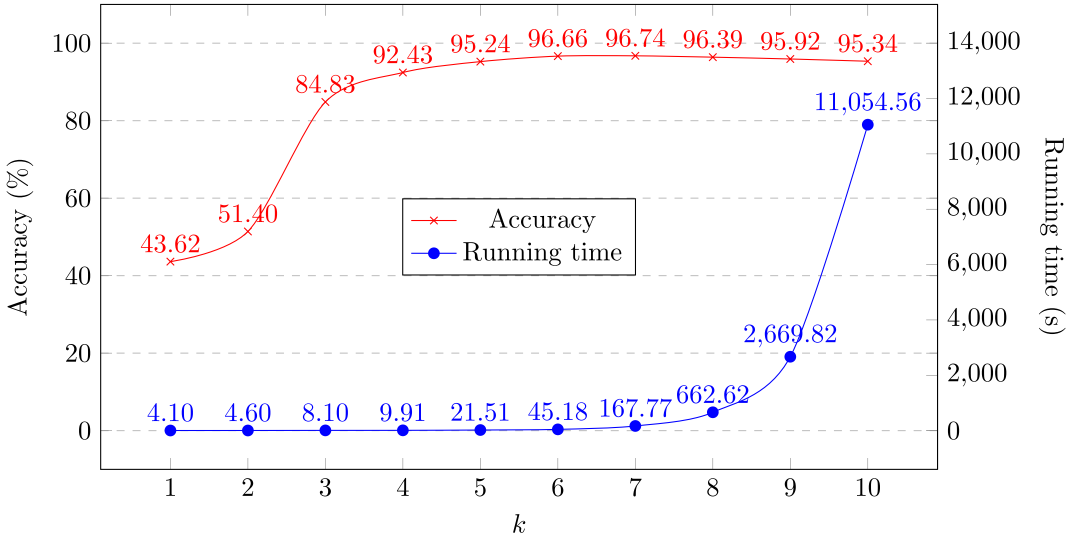


Table 2. Accuracy scores and running times for each of the fifteen classifiers at $k = 6$, for the full set of 6625 whole HIV-1 genomes from the LANL database.

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

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 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 recombinants), 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 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. 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. [91], and the four fastest tools for all datasets together (see Table 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 programs 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. Classification accuracies for all tested HIV-1 subtyping tools, for each testing dataset; average accuracy both with and without weighting datasets by the number of sequences they contain.

Source	Kameris	COMET	CASTOR	SCUEAL	REGA
Nadai (2009) [82]	100.0%	100.0%	81.8%	92.4%	86.4%
Niculescu (2015) [83]	95.9%	96.9%	75.3%	94.8%	100.0%
Paraschiv (2017) [84]	91.9%	73.3%[1]	46.5%	68.6%	87.2%
Rhee (2017) [85]	94.0%	95.4%	0.4%	75.9%	12.8%[2]
Sukasem (2007) [86]	90.0%	91.0%	0.9%	64.3%	8.1%[2]
Eshleman (2001) [87]	88.5%	90.6%	4.2%	84.4%	90.6%
Ssemwanga (2012) [88]	88.3%	90.0%	0.0%	73.3%	95.0%
Wolf (2017) [89]	99.8%	99.8%	61.1%	99.3%	98.2%
TenoRes Study Group (2016) [90]	99.0%	99.0%	28.4%	99.0%	100.0%
van Zyl (2017) [91]	94.9%	93.2%	57.6%	93.2%	94.9%
Huang (2003) [92]	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.

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 frequency vectors (see Methods section), for 9 different pure subtypes or groups (Figure 2), and just subtypes A, B, and C (Figure 3). As can be seen, based on these distances, the points in the plots are grouped

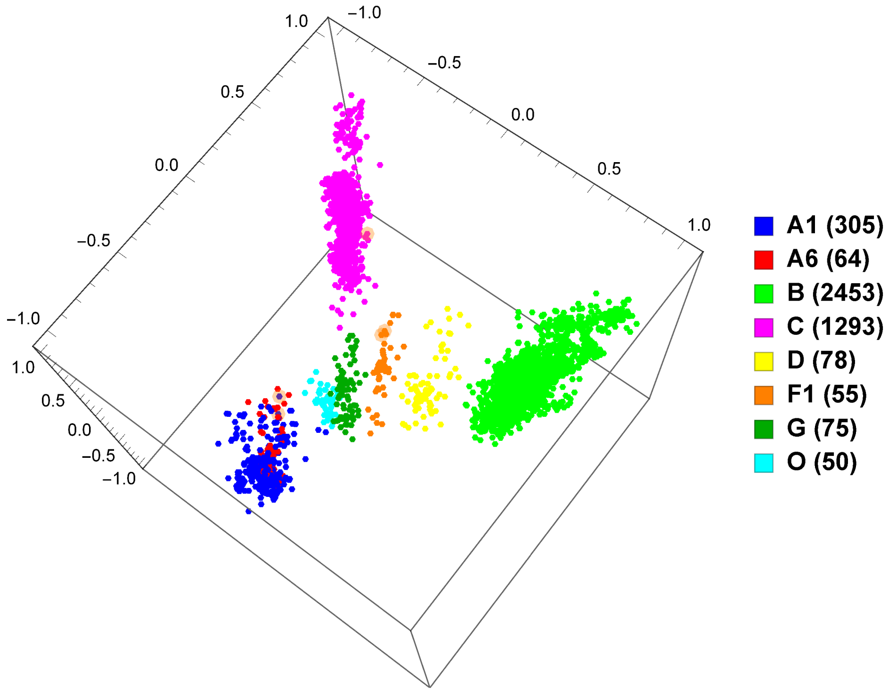
Table 4. Approximate running times for all tested subtyping tools, for the dataset of van Zyl et al. [91] and all datasets listed in Table 3. The van Zyl dataset was chosen at random for this purpose.

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

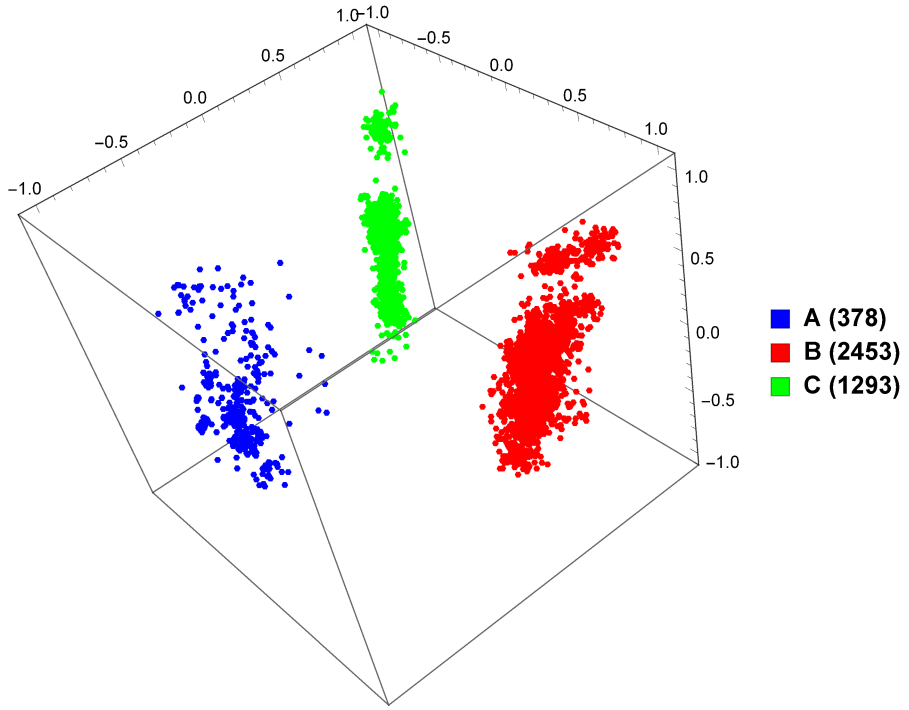
according to known subtypes, and indeed it can be seen that subtypes A1 and A6 group together, and as well B and D group together, as could be expected.

Fig 2. MoDMap of 5686 full-length HIV-1 genomes of 9 different pure subtypes or groups, 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

Fig 3. MoDMap of 5451 full-length HIV-1 genomes of subtypes A, B, and C, at $k = 6$.

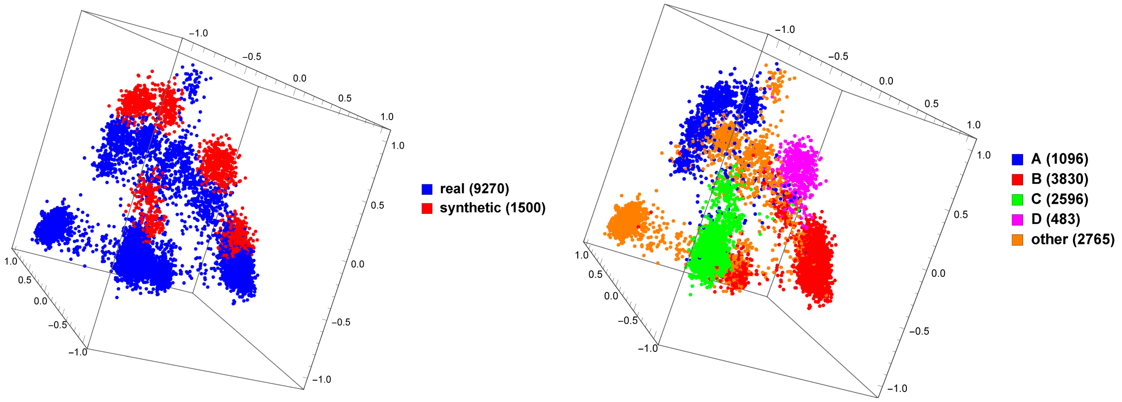


natural from synthetic sequences with perfect accuracy. This suggests that synthetic sequence data should be used with caution, since this result indicates it may not be perfectly representative of natural sequence data – specifically, our result suggests there is some characteristic of the synthetic 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 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 frequency 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.

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 [100] and COMET [8] 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

Fig 4. MoDMap of 9024 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.



webserver hosting the first two major releases of the REGA subtyping algorithm [100] 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 [9] and COMET [8] 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" [101]. 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.

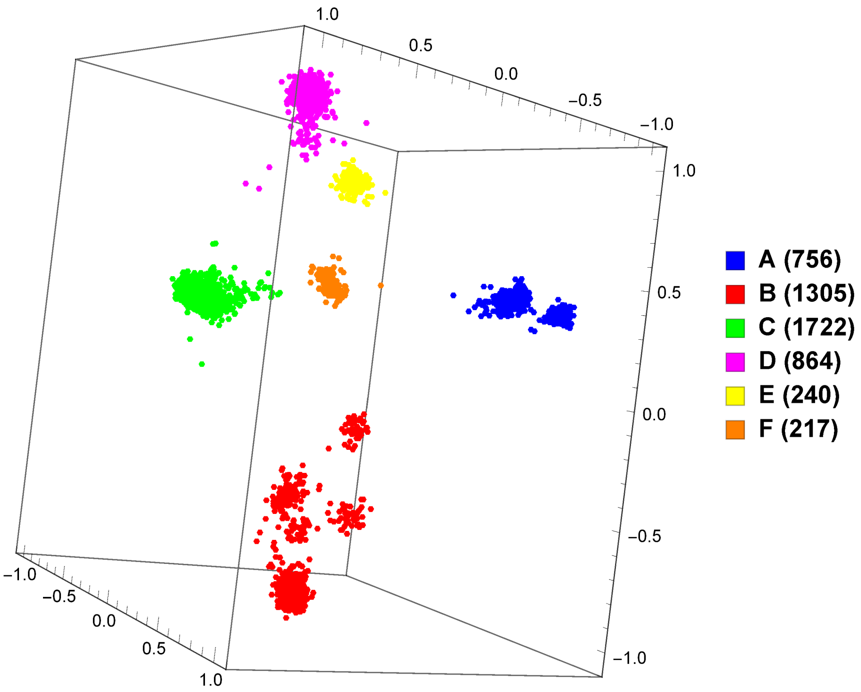
We now discuss some limitations of our approach. Like many machine learning approaches, our method does not provide an accessible explanation as to why a DNA sequence is classified a certain way, compared to a more traditional alignment-based method. In some sense, the classifiers act more as a black 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$_{\text{CPZ}}$ (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$_{\text{CPZ}}$ 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[4]. 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 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.

Fig 5. MoDMap of 5104 whole hepatitis B genomes of 6 different pure subtypes.

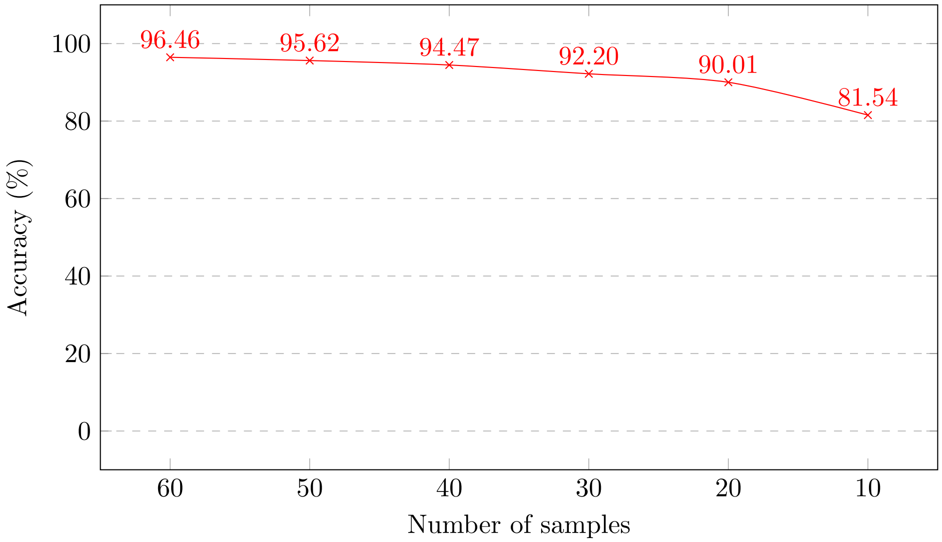


In all the experiments presented above, we use whole assembled genomes or gene sequences. However, next-generation sequencing (NGS) technologies produce as output short reads, often of length 150 to 300 base pairs, and computationally-intensive assembly is required to produce contiguous sequences. Usefully, 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

[4] 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 [5]. However, practical but anecdotal experience suggests subtype U sequences are typically basal.

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, with the results shown in Figure 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.

Fig 6. Classification accuracy scores for the HIV-1 simulated NGS read experiment, with different numbers of samples ("reads" 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 [102] 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.

Supporting information

This research was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) Discovery Grant R2824A01 to L.K., and by grants from the Canadian Institutes of Health Research (CIHR PJT-155990 and PJT-156178) and the Government of Canada through Genome Canada and the Ontario Genomics Institute (OGI-131) to A.P.

Acknowledgments

We wish to thank David Nguyen for his contributions to running preliminary datasets of simulated HIV sequence data sets through the web-based subtyping methods, Gurjit Randhawa for helpful technical and implementation-related discussions, and Dr. Stephen Watt for suggesting the name Kameris for this software.

References

1. Simmonds P, Bukh J, Combet C, Deléage G, Enomoto N, Feinstone S, et al. Consensus proposals for a unified system of nomenclature of hepatitis C virus genotypes. Hepatology. 2005;42(4):962–73.
2. Taylor BS, Sobieszczyk ME, McCutchan FE, Hammer SM. The challenge of HIV-1 subtype diversity. New England Journal of Medicine. 2008;358(15):1590–1602.
3. Worobey M, Gemmel M, Teuwen DE, Haselkorn T, Kunstman K, Bunce M, et al. Direct evidence of extensive diversity of HIV-1 in Kinshasa by 1960. Nature. 2008;455(7213):661–4.
4. Joy JB, Liang RH, Nguyen T, McCloskey RM, Poon AF. Origin and evolution of Human Immunodeficiency Viruses. In: Global Virology I-Identifying and Investigating Viral Diseases. Springer; 2015. p. 587–611.
5. Robertson D, Anderson J, Bradac J, Carr J, Foley B, Funkhouser R, et al. HIV-1 nomenclature proposal. Science. 2000;288(5463):55–55.
6. Clumeck N, Pozniak A, Raffi F. European AIDS Clinical Society (EACS) guidelines for the clinical management and treatment of HIV-infected adults. HIV Medicine. 2008;9(2):65–71.
7. Hirsch MS, Günthard HF, Schapiro JM, Vézinet FB, Clotet B, Hammer SM, et al. Antiretroviral drug resistance testing in adult HIV-1 infection: 2008 recommendations of an International AIDS Society-USA panel. Clinical Infectious Diseases. 2008;47(2):266–285.
8. Struck D, Lawyer G, Ternes AM, Schmit JC, Bercoff DP. COMET: adaptive context-based modeling for ultrafast HIV-1 subtype identification. Nucleic Acids Research. 2014;42(18):e144–e144.
9. Pineda-Peña AC, Faria NR, Imbrechts S, Libin P, Abecasis AB, Deforche K, 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. Infection, Genetics and Evolution. 2013;19:337–348.
10. Pond SLK, Posada D, Stawiski E, Chappey C, Poon AF, Hughes G, et al. An evolutionary model-based algorithm for accurate phylogenetic breakpoint mapping and subtype prediction in HIV-1. PLoS Computational Biology. 2009;5(11):e1000581.
11. Kuiken C, Foley B, Leitner T, Apetrei C, Hahn B, Mizrachi I, et al. HIV sequence compendium 2010. Los Alamos National Laboratory (LANL), Los Alamos, NM (United States); 2010.
12. Gale CV, Myers R, Tedder RS, Williams IG, Kellam P. Development of a novel human immunodeficiency virus type 1 subtyping tool, Subtype Analyzer (STAR): analysis of subtype distribution in London. AIDS Research and Human Retroviruses. 2004;20(5):457–464.
13. Schultz AK, Zhang M, Leitner T, Kuiken C, Korber B, Morgenstern B, et al. A jumping profile Hidden Markov Model and applications to recombination sites in HIV and HCV genomes. BMC Bioinformatics. 2006;7(1):265.

14. Dwivedi SK, Sengupta S. Classification of HIV-1 sequences using profile Hidden Markov Models. PLoS One. 2012;7(5):e36566.

15. Rozanov M, Plikat U, Chappey C, Kochergin A, Tatusova T. A web-based genotyping resource for viral sequences. Nucleic Acids Research. 2004;32(suppl_2):W654–W659.

16. Salminen MO, Carr JK, Burke DS, McCutchan FE. Identification of breakpoints in intergenotypic recombinants of HIV type 1 by bootscanning. AIDS Research and Human Retroviruses. 1995;11(11):1423–1425.

17. Liu Z, Meng J, Sun X. A novel feature-based method for whole genome phylogenetic analysis without alignment: application to HEV genotyping and subtyping. Biochemical and Biophysical Research Communications. 2008;368(2):223–230.

18. Yu C, Hernandez T, Zheng H, Yau SC, Huang HH, He RL, et al. Real time classification of viruses in 12 dimensions. PLoS One. 2013;8(5).

19. Remita MA, Halioui A, Diouara AAM, Daigle B, Kiani G, Diallo AB. A machine learning approach for viral genome classification. BMC Bioinformatics. 2017;18(208).

20. Huang HH, Yu C, Zheng H, Hernandez T, Yau SC, He RL, et al. Global comparison of multiple-segmented viruses in 12-dimensional genome space. Molecular Phylogenetics and Evolution. 2014;81(Supplement C):29 − 36.

21. Li Y, Tian K, Yin C, He RL, Yau SST. Virus classification in 60-dimensional protein space. Molecular Phylogenetics and Evolution. 2016;99(Supplement C):53 − 62.

22. Adetiba E, Olugbara OO, Taiwo TB. Identification of pathogenic viruses using genomic cepstral coefficients with radial basis function neural network. In: Pillay N, Engelbrecht AP, Abraham A, du Plessis MC, Snášel V, Muda AK, editors. Proceedings of the 7th World Congress on Nature and Biologically Inspired Computing (NaBIC2015). Springer International Publishing; 2016. p. 281–291.

23. Wang JD. Comparing virus classification using genomic materials according to different taxonomic levels. Journal of Bioinformatics and Computational Biology. 2013;11(06):1343003.

24. Blaisdell BE. A measure of the similarity of sets of sequences not requiring sequence alignment. Proceedings of the National Academy of Sciences of the United States of America. 1986;83(14):5155–5159.

25. Blaisdell BE. Effectiveness of measures requiring and not requiring prior sequence alignment for estimating the dissimilarity of natural sequences. Journal of Molecular Evolution. 1989;29(6):526–537.

26. Burge C, Campbell AM, Karlin S. Over- and under-representation of short oligonucleotides in DNA sequences. Proceedings of the National Academy of Sciences of the United States of America. 1992;89(4):1358–1362.

27. Karlin S, Ladunga I, Blaisdell BE. Heterogeneity of genomes: measures and values. Proceedings of the National Academy of Sciences of the United States of America. 1994;91(26):12837–12841.

28. Karlin S, Burge C. Dinucleotide relative abundance extremes: a genomic signature. Trends in Genetics. 1995;11(7):283–290.

29. Gelfand MS, Koonin EV. Avoidance of palindromic words in bacterial and archaeal genomes: A close connection with restriction enzymes. Nucleic Acids Research. 1997;25(12):2430–2439.

30. Karlin S, Mrázek J, Campbell AM. Compositional biases of bacterial genomes and evolutionary implications. Journal of Bacteriology. 1997;179(12):3899–3913.

31. Karlin S, Ladunga I. Comparisons of eukaryotic genomic sequences. Proceedings of the National Academy of Sciences of the United States of America. 1994;91(26):12832–12836.

32. Campbell AM, Mrázek J, Karlin S. Genome signature comparisons among prokaryote, plasmid, and mitochondrial DNA. Proceedings of the National Academy of Sciences of the United States of America. 1999;96(16):9184–9189.

33. Wu TJ, Hsieh YH, Li LA. Statistical measures of DNA sequence dissimilarity under Markov chain models of base composition. Biometrics. 2001;57(2):441–448.

34. Shedlock AM, Botka CW, Zhao S, Shetty J, Zhang T, Liu JS, et al. Phylogenomics of nonavian reptiles and the structure of the ancestral amniote genome. Proceedings of the National Academy of Sciences of the United States of America. 2007;104(8):2767–2772.

35. Sims GE, Jun SR, Wu GA, Kim SH. Alignment-free genome comparison with feature frequency profiles (FFP) and optimal resolutions. Proceedings of the National Academy of Sciences of the United States of America. 2009;106(8):2677–2682.

36. Sims GE, Kim SH. Whole-genome phylogeny of Escherichia coli/Shigella group by feature frequency profiles (FFPs). Proceedings of the National Academy of Sciences of the United States of America. 2011;108(20):8329–8334.

37. Wu TJ, Huang YH, Li LA. Optimal word sizes for dissimilarity measures and estimation of the degree of dissimilarity between DNA sequences. Bioinformatics. 2005;21(22):4125–4132.

38. Dai Q, Yang Y, Wang T. Markov model plus k-word distributions: A synergy that produces novel statistical measures for sequence comparison. Bioinformatics. 2008;24(20):2296–2302.

39. Haubold B. Alignment-free phylogenetics and population genetics. Briefings in Bioinformatics. 2014;15(3):407–18.

40. Karamichalis R, Kari L, Konstantinidis S, Kopecki S. An investigation into inter- and intragenomic variations of graphic genomic signatures. BMC Bioinformatics. 2015;16(1):246.

41. Vinga S, Almeida JS. Alignment-free sequence comparison – A review. Bioinformatics. 2003;19(4):513–523.

42. Nalbantoglu OU, Sayood K. Computational genomic signatures. Synthesis Lectures on Biomedical Engineering. 2011;6(2):1–129.

43. Bonham-Carter O, Steele J, Bastola D. Alignment-free genetic sequence comparisons: a review of recent approaches by word analysis. Briefings in Bioinformatics. 2013;15(6):890–905.

44. Zielezinski A, Vinga S, Almeida J, Karlowski WM. Alignment-free sequence comparison: benefits, applications, and tools. Genome Biology. 2017;18(186).

45. Wu GA, Jun SR, Sims GE, Kim SH. Whole-proteome phylogeny of large dsDNA virus families by an alignment-free method. Proceedings of the National Academy of Sciences of the United States of America. 2009;106(31):12826–12831.

46. Kolekar P, Kale M, Kulkarni-Kale U. Alignment-free distance measure based on return time distribution for sequence analysis: Applications to clustering, molecular phylogeny and subtyping. Molecular Phylogenetics and Evolution. 2012;65(2):510–522.

47. Pandit A, Dasanna AK, Sinha S. Multifractal analysis of HIV-1 genomes. Molecular Phylogenetics and Evolution. 2012;62(2):756–763.

48. Pandit A, Vadlamudi J, Sinha S. Analysis of dinucleotide signatures in HIV-1 subtype B genomes. Journal of Genetics. 2013;92(3):403–412.

49. Fiscon G, Weitschek E, Cella E, Presti AL, Giovanetti M, Babakir-Mina M, et al. MISSEL: a method to identify a large number of small species-specific genomic subsequences and its application to viruses classification. BioData Mining. 2016;9(38).

50. Tanchotsrinon W, Lursinsap C, Poovorawan Y. A high performance prediction of HPV genotypes by Chaos game representation and singular value decomposition. BMC Bioinformatics. 2015;16(1).

51. Tanchotsrinon W, Lursinsap C, Poovorawan Y. An efficient prediction of HPV genotypes from partial coding sequences by Chaos Game Representation and fuzzy k-nearest neighbor technique. Current Bioinformatics. 2017;12(5):431–440.

52. Weitschek E, Cunial F, Felici G. LAF: Logic Alignment Free and its application to bacterial genomes classification. BioData Mining. 2015;8(39).

53. Nair VV, Nair AS. 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. p. 35:1–35:8.

54. Nair VV, Mallya A, Sebastian B, Elizabeth I, Nair AS. 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. p. 302–309.

55. Nair VV, S NN, S V, Thushana YS. Texture features from Chaos Game Representation images of genomes. International Journal of Image Processing. 2013;7(2):183–190.

56. Nair VV, Vijayan K, Gopinath DP, Nair AS. ANN based classification of unknown genome fragments using Chaos Game Representation. In: Second International Conference on Machine Learning and Computing (ICMLC 2010). IEEE; 2010. p. 81–85.

57. Golub GH, Van Loan CF. Matrix computations. vol. 3. JHU Press; 2012.

58. Altman NS. An introduction to kernel and nearest-neighbor nonparametric regression. The American Statistician. 1992;46(3):175–185.

59. Tibshirani R, Hastie T, Narasimhan B, Chu G. Diagnosis of multiple cancer types by shrunken centroids of gene expression. Proceedings of the National Academy of Sciences of the United States of America. 2002;99(10):6567–6572.

60. Bishop C. 4.3.4: Multiclass logistic regression. In: Pattern recognition and machine learning. Springer-Verlag New York; 2006. p. 209–210.

61. Cristianini N, Shawe-Taylor J. An introduction to support vector machines and other kernel-based learning methods. Cambridge University Press; 2000.

62. Zhang T. 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. p. 919–926.

63. Breiman L, Friedman J, Olshen R, Stone C. Classification and regression trees. Wadsworth Statistics/Probability. Chapman and Hall; 1984.

64. Breiman L. Random forests. Machine Learning. 2001;45(1):5–32.

65. Freund Y, Schapire RE. A decision-theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Sciences. 1997;55(1):119–139.

66. Hastie T, Rosset S, Zhu J, Zou H. Multi-class AdaBoost. Statistics and its Interface. 2009;2(3):349–360.

67. Chan TF, Golub GH, LeVeque RJ. Updating formulae and a pairwise algorithm for computing sample variances. In: COMPSTAT 5th Symposium. Springer; 1982. p. 30–41.

68. Friedman J, Hastie T, Tibshirani R. 4.3: Linear Discriminant Analysis. In: The Elements of Statistical Learning. vol. 1. New York: Springer Series in Statistics; 2001. p. 106–119.

69. Hinton GE. Connectionist learning procedures. Artificial Intelligence. 1989;40(1-3):185–234.

70. Kingma D, Ba J. Adam: A method for stochastic optimization. arXiv preprint arXiv:14126980. 2014;.

71. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research. 2011;12:2825–2830.

72. Refaeilzadeh P, Tang L, Liu H. In: Liu L, Özsu MT, editors. Cross-Validation. Boston, MA: Springer US; 2009. p. 532–538.

73. Kari L, Hill KA, Sayem AS, Karamichalis R, Bryans N, Davis K, et al. Mapping the space of genomic signatures. PLoS One. 2015;10(5).

74. Karamichalis R, Kari L, Konstantinidis S, Kopecki S, Solis-Reyes S. Additive methods for genomic signatures. BMC Bioinformatics. 2016;17(1):313.

75. Krause EF. Taxicab geometry: An adventure in non-Euclidean geometry. Mineola, New York: Courier Dover Publications; 2012.

76. Borg I, Groenen P. Modern Multidimensional Scaling: Theory and Applications. 2nd ed. Springer; 2005.

77. Karamichalis R, Kari L. MoDMaps3D: an interactive webtool for the quantification and 3D visualization of interrelationships in a dataset of DNA sequences. Bioinformatics. 2017;33(19):3091–3093.

78. Jain AK, Chandrasekaran B. 39 dimensionality and sample size considerations in pattern recognition practice. In: Classification Pattern Recognition and Reduction of Dimensionality. vol. 2 of Handbook of Statistics. Elsevier; 1982. p. 835–855.

79. Raudys SJ, Jain AK. Small sample size effects in statistical pattern recognition: Recommendations for practitioners. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1991;13(3):252–264.

80. Tebit DM, Arts EJ. Tracking a century of global expansion and evolution of HIV to drive understanding and to combat disease. The Lancet Infectious Diseases. 2011;11(1):45–56.

81. Leitner T, Korber B, Daniels M, Calef C, Foley B. HIV1 subtype and circulating recombinant form (CRF) reference sequences, 2005. 2005;2005.

82. Nadai Y, Eyzaguirre LM, Sill A, Cleghorn F, Nolte C, Charurat M, et al. HIV-1 epidemic in the Caribbean is dominated by subtype B. PLoS One. 2009;4(3):e4814.

83. Niculescu I, Paraschiv S, Paraskevis D, Abagiu A, Batan I, Banica L, et al. Recent HIV-1 outbreak among intravenous drug users in Romania: evidence for cocirculation of CRF14_BG and subtype F1 strains. AIDS Research and Human Retroviruses. 2015;31(5):488–495.

84. Paraschiv S, Banica L, Nicolae I, Niculescu I, Abagiu A, Jipa R, et al. Epidemic dispersion of HIV and HCV in a population of co-infected Romanian injecting drug users. PLoS One. 2017;12(10):e0185866.

85. Rhee SY, Varghese V, Holmes SP, Van Zyl GU, Steegen K, Boyd MA, et al. Mutational correlates of virological failure in individuals receiving a WHO-recommended tenofovir-containing first-line regimen: An international collaboration. EBioMedicine. 2017;18:225–235.

86. Sukasem C, Churdboonchart V, Chasombat S, Kohreanudom S, Watitpun C, Pasomsub E, et al. Surveillance of genotypic resistance mutations in chronic HIV-1 treated individuals after completion of the National Access to Antiretroviral Program in Thailand. Infection. 2007;35(2):81–88.

87. Eshleman SH, Becker-Pergola G, Deseyve M, Guay LA, Mracna M, Fleming T, 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). The Journal of Infectious Diseases. 2001;184(7):914–917.

88. Ssemwanga D, Kapaata A, Lyagoba F, Magambo B, Nanyonjo M, Mayanja BN, et al. Low drug resistance levels among drug-naive individuals with recent HIV type 1 infection in a rural clinical cohort in southwestern Uganda. AIDS Research and Human Retroviruses. 2012;28(12):1784–1787.

89. Wolf E, Herbeck JT, Van Rompaey S, Kitahata M, Thomas K, Pepper G, et al. Phylogenetic evidence of HIV-1 transmission between adult and adolescent men who have sex with men. AIDS Research and Human Retroviruses. 2017;33(4):318–322.

90. Group TS, 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. The Lancet Infectious Diseases. 2016;16(5):565–575.

91. van Zyl GU, Grobbelaar CJ, Claassen M, Bock P, Preiser W. Moderate levels of preantiretroviral therapy drug resistance in a generalized epidemic: time for better first-line ART? AIDS. 2017;31(17):2387–2391.

92. Huang DD, Giesler TA, Bremer JW. Sequence characterization of the protease and partial reverse transcriptase proteins of the NED panel, an international HIV type 1 subtype reference and standards panel. AIDS Research and Human Retroviruses. 2003;19(4):321–328.

93. Edgar RC. MUSCLE: multiple sequence alignment with high accuracy and high throughput. Nucleic Acids Research. 2004;32(5):1792–1797.

94. Larsson A. AliView: a fast and lightweight alignment viewer and editor for large datasets. Bioinformatics. 2014;30(22):3276–3278.

95. Darriba D, Taboada GL, Doallo R, Posada D. jModelTest 2: more models, new heuristics and parallel computing. Nature Methods. 2012;9(8):772–772.

96. Guindon S, Dufayard JF, Lefort V, Anisimova M, Hordijk W, Gascuel O. New algorithms and methods to estimate maximum-likelihood phylogenies: assessing the performance of PhyML 3.0. Systematic Biology. 2010;59(3):307–321.

97. Rambaut A. FigTree; 2016. Available from: `http://tree.bio.ed.ac.uk/software/figtree/`.

98. Fletcher W, Yang Z. INDELible: a flexible simulator of biological sequence evolution. Molecular Biology and Evolution. 2009;26(8):1879–1888.

99. Poon AF. Phylodynamic inference with kernel ABC and its application to HIV epidemiology. Molecular Biology and Evolution. 2015;32(9):2483–2495.

100. De Oliveira T, Deforche K, Cassol S, Salminen M, Paraskevis D, Seebregts C, et al. An automated genotyping system for analysis of HIV-1 and other microbial sequences. Bioinformatics. 2005;21(19):3797–3800.

101. Aziz N, Zhao Q, Bry L, Driscoll DK, Funke B, Gibson JS, et al. College of American Pathologists' laboratory standards for next-generation sequencing clinical tests. Archives of Pathology and Laboratory Medicine. 2014;139(4):481–493.

102. Asgari E, Mofrad MRK. Continuous Distributed Representation of Biological Sequences for Deep Proteomics and Genomics. PLoS One. 2015;10(11):1–15.