Hsin-Hsiung Huang[1] / Shuai Hao[2] / Saul Alarcon[2] / Jie Yang[2]

Comparisons of classification methods for viral genomes and protein families using alignment-free vectorization

[1] Department of Statistics, University of Central Florida, 4000 Central Florida Blvd, Orlando, FL 32816, USA, E-mail: hsin.huang@ucf.edu
[2] Department of Mathematics, Statistics, and Computer Science, University of Illinois at Chicago, Chicago, IL, USA

Abstract:
In this paper, we propose a statistical classification method based on discriminant analysis using the first and second moments of positions of each nucleotide of the genome sequences as features, and compare its performances with other classification methods as well as natural vector for comparative genomic analysis. We examine the normality of the proposed features. The statistical classification models used including linear discriminant analysis, quadratic discriminant analysis, diagonal linear discriminant analysis, *k*-nearest-neighbor classifier, logistic regression, support vector machines, and classification trees. All these classifiers are tested on a viral genome dataset and a protein dataset for predicting viral Baltimore labels, viral family labels, and protein family labels.

Keywords: viral genomes, protein, family labels, Natural Vector, statistical classification models

DOI: 10.1515/sagmb-2018-0004

1 Introduction

Virus classification is one of the prevailing approaches in which virologists study the various diversity of viruses. Typically, viruses are grouped by their nucleic acid genome characteristics. There are two main classification systems, Baltimore classification and International Committee on Taxonomy of Viruses (ICTV) system. Baltimore classification was first proposed by David Baltimore, a Nobel laureate, (Baltimore, 1971). Viruses are assigned to seven Baltimore groups (I, II, III, IV, V, VI, VII) according to their type of genome and mechanism of mRNA reproduction. ICTV classification is a hierarchical system. It consists of five levels of classes: order, family, subfamily, genus, species. The names of viral taxa in a specific level share a unique suffix. For instance, all names of order end with suffix "-virales."

In order to classify a new virus, one needs quantify the similarity between the given virus and known viruses via comparing their genome sequences. The subsequent classification result is based on the sequence comparisons. Two categories of sequence comparisons are widely accepted and well developed, Sequence alignment and alignment-free vectorization.

The widely used sequence alignment assumes the conservation of contiguity of homologous segments. Pairwise sequence alignment (PSA) returns a score of similarity between two sequences according to a proposed "score system." In the case where there are multiple sequences, one can generalize PSA to Multiple Pairwise Alignment (MSA) which optimize the alignment scores of all pairwise comparisons. Each time a new virus is given, one needs to repeat the optimization procedure to obtain the alignment score. Therefore the computational workload escalates as a power function of the length of a sequence (Vinga & Almeida, 2003).

Conversely, alignment-free methods typically does not require intensive computation (Sims et al. 2009). There are two classes of alignment-free methods. The first type is to convert sequences to an summarized representation upon which subsequent analysis, are made (Vinga, 2007). The other type is to transform sequences to numerical vectors based on word frequency, such as k-mers (Ghor et al. 2009), composition vectors (Chan, Wang & Yeung , 2010), and natural vectors (Deng et al., 2011; Yu et al., 2013; Huang et al., 2014b). The similarity between sequences is assessed by measuring well-defined distance in vector space. One of its merits is linear algebra and statistical theory are available. However, there is an obvious bias that alignment-free vectorization ignores spacial structure of genome sequences by simply treating them as strings (Weitschek, Cunial & Felici, 2015). Recently, researchers proposed new approaches include integrating various types of alignment-free

Hsin-Hsiung Huang is the corresponding author.

methods (Huang et al., 2011; Huang, 2016), combining alignment and alignment-free methods in a sequential manner (Huang et al., 2016), and using Fourier or wavelets to transform genome sequences (Hoang et al., 2015; Huang & Girimurugan, 2018), so that subjects could be well classified.

While a genomics dataset is enormously huge, statistical learning methods can be applied to alignment-free vector representation of the genome sequences. However, it is noteworthy that performances of classifiers are hardly the same upon various situations. Dudoit, Fridlyand, and Speed (2002) compared the performances of discriminant methods, such as linear discriminant analysis, k-nearest-neighbor and classification trees, and other machine learning techniques including bagging and boosting, on three gene expression datasets in classifying tumor subtypes. Huang, Xu, and Yang (2014a) implemented logistic regression, support vector machine, and permanental classification methods on genome-wide association study data in predicting hypertension.

In the field of protein classification, cases are very similar to those of viruses. Alignment-free methods are applicable to amino acids sequences as well. It falls in the framework of virus classification with minor modification (Huang, Xu & Yang , 2014a; Huang, 2014). Note that the classifiers considered in literature are more complicated and time consuming. For example, the bagging and boosting are more powerful than a single classifiers. Neural Networks have remarkable applications and turns out work well on various problems. However, they requires more training time and some of them lack statistical or rational interpretations.

In statistics, linear discriminant analysis and quadratic discriminant analysis are both simple and popular classification methods when features are normally distributed. In this paper, we proposed an approach of vectorization that derived from the Natural Vector, in which the features approximately follow Normal distributions, and seven classification methods are used for testing on the proposed features obtained from the virus and protein dataset. Meanwhile, performances of proposed vectors are compared with those of natural vectors. It is shown that combination of the proposed vector and simple classifiers, like LDA or QDA, offers a faster alternative to predicting family labels for both virus and proteins with minor trade-off on accuracy. The significance lies in the case where even a researcher equipped with normal computers can handle large classification tasks within a reasonable period of time considering the fact that high performance computing resources are usually scarce.

The rest of the paper is organized as follows. Section 2 introduces proposed alignment-free vectorization classification methods and other classifiers, and implementation details are in Section 3. Section 4 provides a summary and discussion.

2 Methods

2.1 Alignment-free vectorization methods

Viral genome sequence usually refers to DNA/RNA sequence. A DNA/RNA sequence which consists of four kinds of nucleotides, (A)denine, (C)ytosine, (G)uanine and (T)hymine (or (U)racil in RNA) is represented as a string of letters A, C, G, $T(U)$. There are several alignment-free methods for genome comparisons that have been considered in literature, including k-mer, Natural Vector (NV), Composite Vector (CV) and Q-Vector (QV) (Hernandez & Yang, 2013). Additionally, all these alignment-free methods are applicable to protein sequences. A protein sequence which is a chain composed of 20 possible amino acids is expressed as a string of 20 letters.

2.1.1 Natural vector

The recently developed Natural Vector representation has be successfully applied for clustering and classifying genome datasets (Deng et al., 2011; Yu et al., 2013; Huang et al., 2014b). Given a genome sequence, this method converts a genome sequence into a 12-dimensional vector by calculating the total counts, mean positions, and variance of positions for each type of nucleotide. To be more specific, for a sequence of n letters, $u = (u_1, u_2, ..., u_n)$, where u_i is from $\{A, C, G, T\}$, we define an indicator function

$$I_k(x) = \begin{cases} 0 & x \neq k \\ 1 & x = k \end{cases}, \tag{1}$$

where $k \in \{A, C, T, G\}$. Then

$$n_k = \sum_{i=1}^{n} I_k(u_i), \text{ the total count of letter } k \tag{2}$$

$$\mu_k = \sum_{i=1}^{n} \frac{i \cdot I_k(u_i)}{n_k}, \text{ the mean position of letter } k \tag{3}$$

$$D_k = \sum_{i=1}^{n} \frac{(i-\mu_k)^2 I_k(u_i)}{n_k n}, \text{ the normalized variance of position of letter } k. \tag{4}$$

The 12-dimensional natural vector is

$$(n_A, \mu_A, D_A, n_C, \mu_C, D_C, n_G, \mu_G, D_G, n_T, \mu_T, D_T) \tag{5}$$

2.2 Proposed method

Suppose there are K classes and each with n_i members. if n_i is large enough, multivariate central limit theorem (Darling, 1975) yields the following result,

Theorem 2.1: (Multivariate Central Limit Theorem)
Given a class k, suppose that $X = (x_1, ..., x_p)$ is a random vector in $\mathbb{R}^p$ with mean $\mathbb{E}_k[X_i]$ and covariance Σ_k. We assumed that $\mathbb{E}_k[x_i^2] < \infty$. If $X_1, X_2, ...$ are identical and independent sequences of X then

$$S_n = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} (X_i - \mathbb{E}_k[X_i]) \xrightarrow{D} N(0, \Sigma), \tag{6}$$

where $\xrightarrow{D}$ means convergence in distribution.

The multivariate central limit theorem leads to a novel genome representation method using the first and second moments of each nucleotide position. Afterwards, Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA), which require normal-distributed features, are applicable to the vectorized genomes. Therefore, we propose the following modified natural vector

$$x = (\mu'_A, D'_A, \mu'_C, D'_C, \mu'_G, D'_G, \mu'_T, D'_T), \tag{7}$$

where

$$\mu'_k = \sum_{i=1}^{n} \frac{i \cdot I_k(u_i)}{\sqrt{n}}, \text{ mean position of letter } k \tag{8}$$

$$D'_k = \sum_{i=1}^{n} \frac{(i-\mu_k)^2 I_k(u_i)}{\sqrt{n}}, \text{ variance of position of letter } k. \tag{9}$$

In the following section, we use quantile-quantile (Q-Q) plots to verify that the modified natural vectors are approximately normal distributed for the investigated datasets.

2.3 Classification methods

Suppose $y = (y_1, ..., y_n)'$ is a vector of group labels, and observation corresponding to i-th observation is $x_i \in \mathbb{R}^p$, $i = 1, 2, \dots, n$. Goal is to classify y_0 given a new observation point x_0. Classification methods have been widely used for genome sequences classification (Maddouri & Elloumi, 2002; Polychronopoulos et al., 2014). The classification methods (Hastie, Tibshirani & Friedman , 2009) are summarized as follows.

2.3.1 Discriminant analysis

Quadratic Discriminant Analysis (QDA) assumes features in each class k following a multivariate Normal distribution, with mean vector μ_k and common covariance matrix Σ_k, $k = 1, \ldots, K$. The discriminant score is

$$\delta_k = -\frac{1}{2}\log\det(\Sigma_k) - (x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) + \log\pi_k, \tag{10}$$

where π_k is prior probability of class k. The classification result is

$$y_0 = \arg\max_k \delta_k. \tag{11}$$

If the within class covariance matrix for all classes are identical and diagonal, the discriminant classifier becomes Diagonal Linear Discriminant Analysis (DLDA). i.e. observations in the same class are independent. When the within class covariance matrix for all classes are identical, the discriminant classifier is Linear Discriminant Analysis (LDA).

2.3.2 k-nearest-neighbor (KNN)

The k-nearest-neighbor (knn) is the simplest and most intuitive classifier. Suppose there is a new point x_0 and known points $\{(x_i, y_i)\}_{i=1}^n$, knn calculates the mutual distances between x_0 and all points, and assigns x_0 to a class via majority vote of its k-nearest neighbors. Different distances are developed for various purposes, and in this paper, Euclidean distance is chosen. k, the number of adjacent neighbors under consideration, is the only tuning parameter which could be tuned by cross-validation.

2.3.3 Logistic regression

Logistic regression classifier models the transformation of probability or the log-odds of an single observation belonging to a certain class. Suppose there are K classes. The following logistic models are fitted to a dataset $\{(x_i, y_i)\}_{i=1}^n$.

$$logit[P(y = k|X = x)] = \alpha_{0k} + x\beta_k, \tag{12}$$

where $k = 1, 2, \ldots, K$. The probability that a new point x_0 belongs to jth class is

$$P(y_0 = j|X = x_0) = \frac{\exp(\alpha_j + x_0\beta_j)}{\sum_{j=1}^K \exp(\alpha_j + x_0\beta_j)}. \tag{13}$$

The predicted class is the one with the largest probability.

2.3.4 Support vector machines (SVM)

Once the observation points are mapped into points in space, support vector machines (SVM) (Cortes & Vapnik, 1995) seeks to find a hyperplane that has largest margin to separate points from two different classes. Its theoretical foundation is designated for two-class case, and it is a non-linear classifier.

Suppose the class labels are represented by $y_i \in \{1, -1\}$, and $x \in \mathsf{R}^p$. The computational target of SVM is to optimize

$$\min_{\beta,\beta_0} \frac{1}{2}||\beta||^2 + C\sum_{i=1}^n \xi_i \tag{14}$$

$$\text{subject to } \xi_i \geq 0, y_i(x_i^T\beta + \beta_0) \geq 1 - \xi_i, \forall i, \tag{15}$$

where β and β_0 are parameters of decision boundary, ξ_i's are slack variables, and C is the cost tuning parameter. In solving the nonlinear boundary classification problem, a kernel is introduced (Huang, Xu & Yang , 2014a 2014a). Here we choose the radial basis function (RBF) kernel,

$$K(x, x')\ \exp(-\gamma\|x - x'\|^2), \tag{16}$$

where γ is the tuning parameter for the RBF kernel. The tuning parameters are tuned by cross validation before prediction is performed.

It is noteworthy that SVM is born to tackling two-class problems, and one needs to generalize it to multiple-class situations. In this paper, we use a one-against-one approach, in which there are $K(K-1)/2$ classifiers will be trained and tested if there are K classes. The classification result is delivered by a voting scheme.

2.3.5 Classification Trees

Classification Trees recursively partitions the space of input features into subspaces until some stopping rule has been reached. The resulting partition is the corresponding classifier. To 'grow' a tree, we use Classification and Regression Trees (CART) method, and Gini Index is chosen to be the measure of impurity measure of each partition. Now we are ready to present the results of examples upon which the proposed methods are tested.

3 Results

The aforementioned classifiers are tested on proposed modified natural vectors. Vectors are generated from two datasets, virus reference sequences, and homo-protein sequences. On the viruses dataset, it includes both Baltimore labels and family labels. Therefore are three examples tested, virus Baltimore labels, virus family labels, and protein family labels.

3.1 Accessing normality

First of all, the multivariate normality assumption is carefully checked prior to the training and testing procedure. The multivariate normality assumption is the key to proposed vectors. However, tools for its justification are limited because it is a multi-dimensional problem. Several statistical tests are available in literature, see Selcuk, Dincer, and Gokmen (2016) for a short review. One of the drawbacks is those tests tend to reject the null hypothesis (multivariate normality assumption). A more straightforward yet easily recognizable approach is via Chi-squared (χ^2) Q-Q plot.

Suppose $x_1, x_2, ..., x_n$ are *i.i.d* random vectors of length p from multivariate normal distribution with mean vector μ and variance-covariance matrix Σ, i.e. $N_p(\mu, \Sigma)$. The plot of squared mahalanobis distances against quantiles of Chi-Square distribution with p degree of freedom, is the Q-Q plot we used to access to the normality. Note that the squared Mahalanobis distance is calculated by

$$d_i^2 = (x_i - \mu)\Sigma^{-1}(x_i - \mu)^T \overset{i.i.d}{\sim} \chi_p^2. \tag{17}$$

In practice, because μ and Σ are usually unknown, the d_i^2's are obtained by plugging in observed mean and variance-covariance matrix.

3.2 Viral genome with Baltimore labels

The virus reference sequence dataset was downloaded from the National Center of Biotechnology Information NCBI (2016). It contains 6009 reference viruses including 4271 single-segmented viruses. The rest are regarding multi-segmented viruses. We excluded those with more than one segments of reference sequences due to its complexity in computation. Table 1 displays the distribution of viruses in seven Baltimore groups. It is noteworthy that three groups, Groups I, II, and IV, are dominating in numbers. Sequences of of DNA/RNA virus are extracted along with other biological information. Although DNA and RNA contains different kinds of nucleotides, in this dataset, sequences of RNA viruses are in terms of A, C, G, and T.

Figure 1 is Chi-squared Q-Q plots of the seven Baltimore classes. It is shown most of the classes are approximately normal, except class 'I'. However, we believe it is not a severe problem and we proceed to the next phase of training and testing.

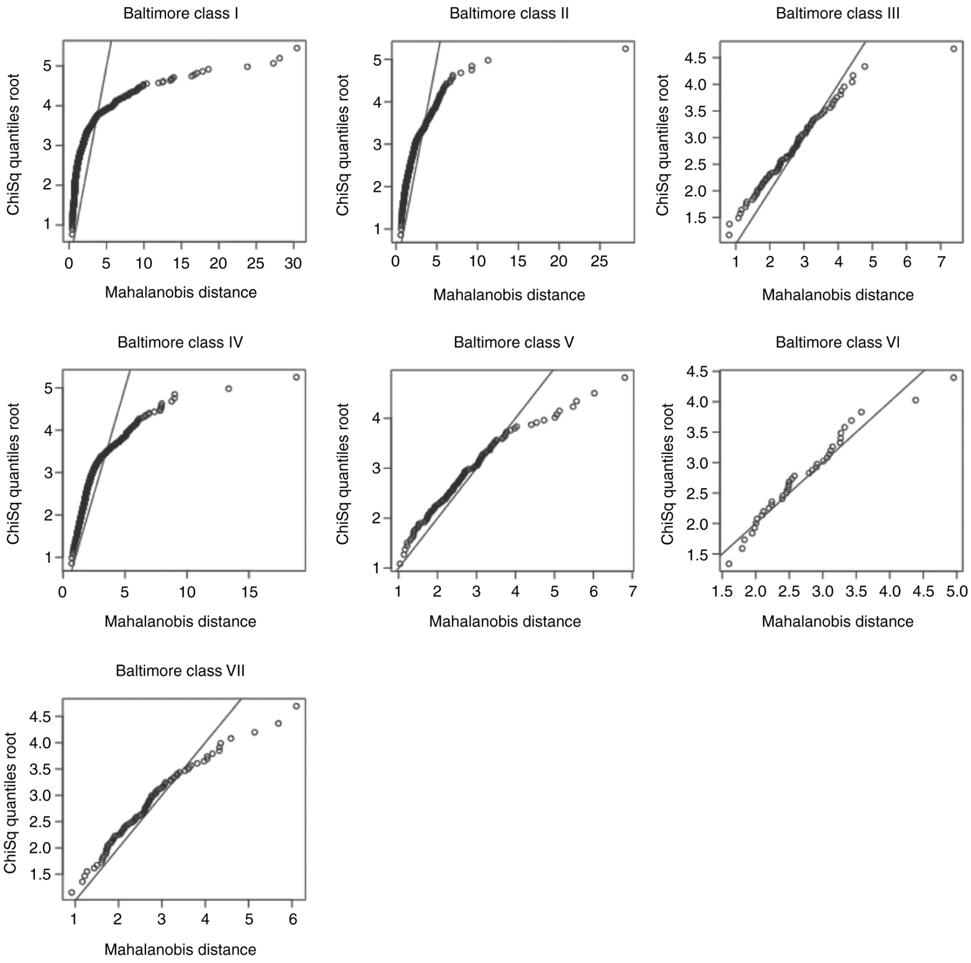

Figure 1: Chi-square Q-Q plots of the Baltimore classes.

3.3 Viral genome with ICTV labels

The virus dataset from NCBI also contains family labels. But some of them do not have assigned family labels. In the next example, we exclude those without family labels. Note that the cardinality of each family are unbalanced and some of them are very small. We chose 12 bigger families with number of members greater than 50. Table 2 displays the distribution of 12 families along with their member counts. In summary, there are 2492 viruses in this example.

Table 1: Distribution of the Baltimore classes.

Group	I	II	III	IV	V	VI	VII
Number of viruses	2099	884	94	893	160	38	103

Table 2: Distribution of viral families.

Family	Circoviridae	Flaviviridae	Geminiviridae	Myoviridae
# of members	132	85	251	400
Family	Picornaviridae	Podoviridae	Polyomaviridae	Potyviridae
# of members	94	257	86	123

Family	Papillomaviridae	Parvoviridae	Rhabdoviridae	Siphoviridae
# of members	133	90	80	761

The normality assumption is tested prior to the model fitting. Figure 2 shows the Chi-square Q-Q plots of the modified natural vectors from the 12 virus families. Except for two families, all others are considered to be normal because there Q-Q plots are approximately diagonal lines.

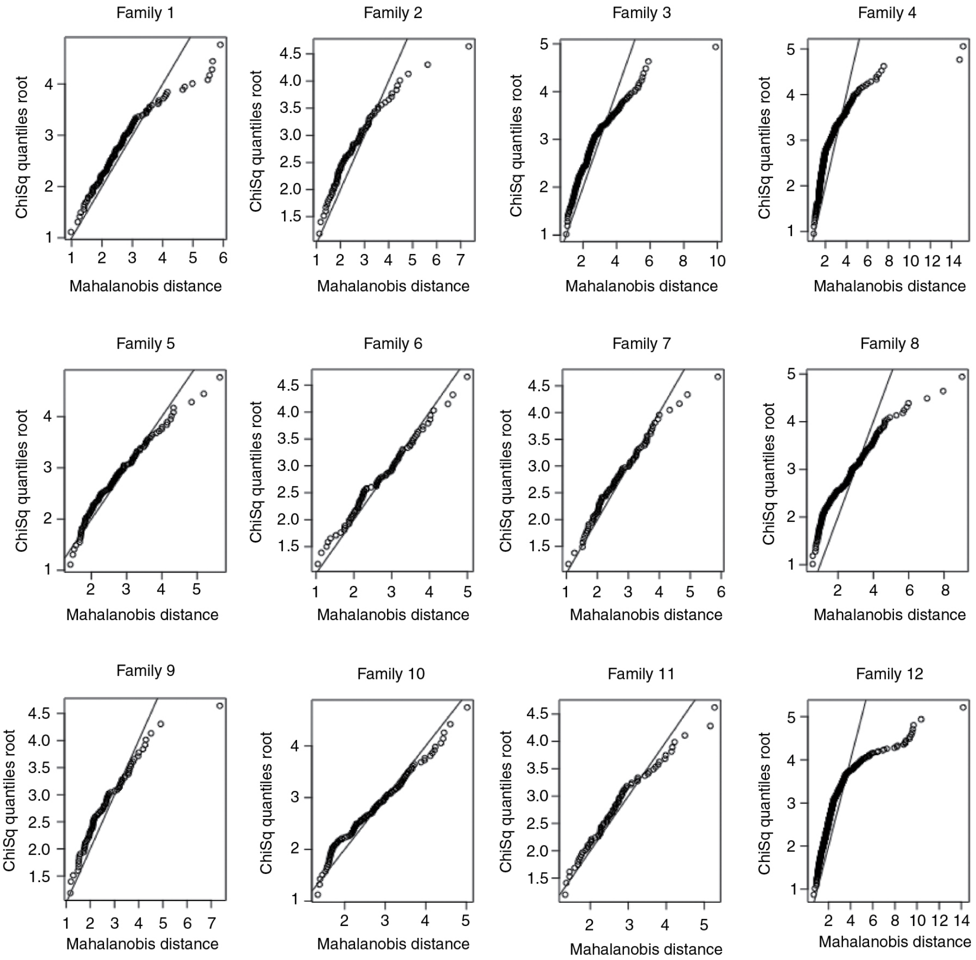

Figure 2: Chi-square Q-Q plots of viral families.

3.4 Protein sequences with family labels

It includes 20199 sequences, 6053 of them have missing family labels NCBI (2016). We choose the biggest 9 families (contains more than 50 members), 1690 samples in total. The distribution of the families are displayed in Table 3. The normality is shown in the Q-Q plot in Figure 3. Six of nine families are nearly normally distributed. Families with Q-Q plots that are not straight diagonal lines are somehow not too far away. Therefore, we are able to proceed to next step, and assume the multivariate normality.

Table 3: Distribution of protein families.

Family	Cytochrome P450	G-protein coupled receptor	TRIM/RBCC
# of members	60	670	64
Family	Mitochondrial carrier	Intermediate filament	Peptidase S1
# of members	52	74	76
Family	MHC class I	Krueppel C2H2-type zinc-finger	Small GTPase
# of members	83	538	73

Table 4: Results of viral Baltimore labels classification. The unit of time is a second (s).

Classifier	AUC	Micro-F	Time (s)
(a) Results of the proposed vector			
LDA	0.5	0.4865	0.06 (0.01)
QDA	0.8316	0.8073	0.07 (0.003)
KNN	0.8257	0.8864	0.18 (0.01)
SVM	0.7893	0.8539	275.0 (1.8)
Logit	0.5004	0.4882	15.0 (0.4)
DLDA	0.6946	0.7398	0.06 (0.01)
Tree	0.7725	0.8373	0.71 (0.02)
(b) Results of Natural Vector			
LDA	0.5157	0.4834	0.09 (0.02)
QDA	0.8577	0.8202	0.10 (0.02)
KNN	0.8851	0.9305	0.23 (0.01)
SVM	0.8406	0.8936	339.8 (3.8)
Logit	0.5050	0.4828	22.0 (0.3)
DLDA	0.6989	0.7232	0.07 (0.01)
Tree	0.7676	0.8441	1.00 (0.03)

Table 5: Results of viral families classification. The unit of time is a second (s).

Classifier	AUC	Micro-F	Time (s)
(a) Results of the proposed vector			
LDA	0.7114	0.4912	0.04 (0.01)
QDA	0.8876	0.7661	0.06 (0.01)
KNN	0.9063	0.8555	0.07 (0.01)
SVM	0.9074	0.8591	226.4 (2.5)
Logit	0.8961	0.7769	44.9 (0.3)
DLDA	0.707	0.4908	0.04 (0.01)
Tree	0.876	0.7608	0.75 (0.02)
(b) Results of Natural Vector			
LDA	0.6952	0.4975	0.06 (0.01)
QDA	0.9106	0.7945	0.07 (0.02)
KNN	0.9379	0.8937	0.09 (0.01)
SVM	0.9376	0.8921	203.4 (1.5)
Logit	0.9076	0.7905	80.0 (0.4)
DLDA	0.7123	0.4498	0.06 (0.01)
Tree	0.8743	0.7869	0.97 (0.02)

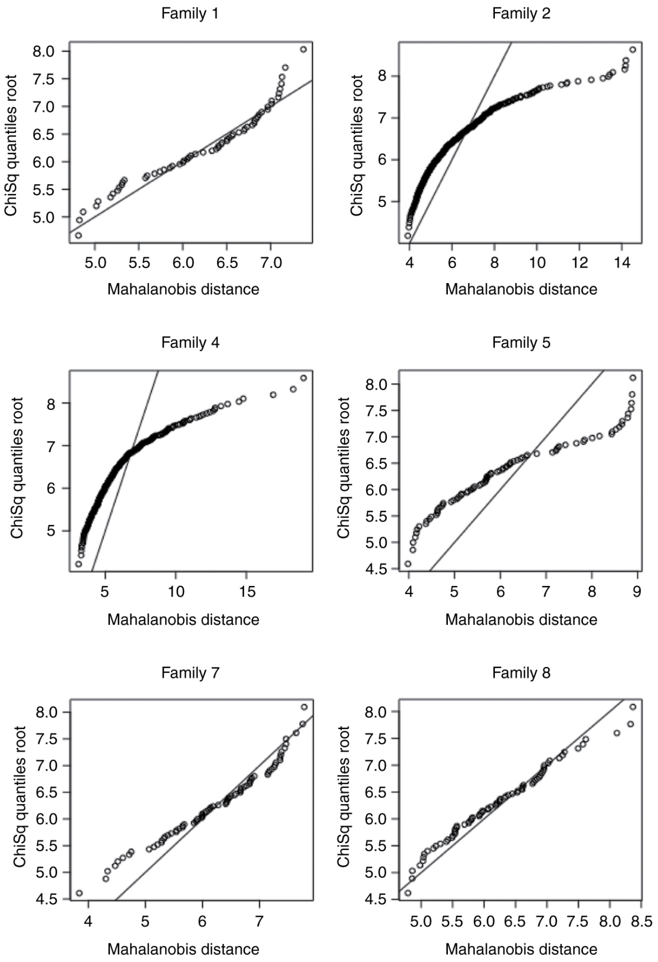

Figure 3: Chi-square Q-Q plots of protein families.

3.5 Data analysis

Classifiers are trained via 5-fold cross validation. Since the data is unbalanced, we randomly partitioned the entire dataset into 5 disjoint subsets while controlling the proportion of viruses in each group unchanged. For a comprehensive review on cross-validation, please read Hastie, Tibshirani & Friedman , 2009.

3.5.1 Performance metrics

Performance metrics are provided for the purpose of comparison of classifiers. Because all examples are unbalanced multi-class classification problems, metrics of classifiers are carefully chosen. They are AUC, micro-F, and computing time. Because of the magnitude of computing time for some classifiers are very small (around 0.1s), we repeat the programs for 100 times for LDA, QDA, KNN, DLDA, and Classification tree and 10 times for logistic regression and support vector machine. Finally average computing time as well as interquartile ranges are reported.

AUC is the area under receiver operating characteristic curve. Original AUC is defined for two classes case, in this paper we used the extension by Hand and Till (2001). It averages all pairwise class AUCs. The micro-*F* measure is a harmonic mean of precision and recall and it equals to accuracy. Note that *F*-measure is the one that combines both precision and recall and its value will close to whichever is smaller. The algorithms are programmed in R and run on an Laptop with 1.4 GHz CPU and 4GB RAM.

3.5.2 Virus Baltimore labels

The performance metrics of both proposed vector and natural vector are summarized in Table 4. In general, performances of classifiers on proposed vector are better than those on natural vector with respect to AUC and

Micro-F, but the differences are very small. For proposed vector (in panel (a)), KNN, SVM, QDA, and classification tree have prediction accuracy greater than 0.8, with the highest 0.8865 from KNN. The QDA depends on the normality assumption, and it turns out QDA also works well in this case with prediction accuracy being 0.8073 which is close to those of other top classifiers. In terms of AUC, which can be interpreted as “the probability that a randomly chosen positive example is ranked higher than a randomly chosen negative example.” (Fawcett, 2006), QDA and nearest neighbor has virtually identical values, while SVM has slightly less AUC value. For natural vector, We have similar observations in panel (b), with highest accuracy being 0.9305 from KNN and 0.8202 for QDA. In terms of AUC, those for QDA and KNN are increased by 0.02 and 0.06 respectively compared to parallel metrics for proposed vector, which is not regarded as significant improvement.

The computation time is also a very important metric. It turns out classifiers on proposed vectors generally cost less time than that on natural vector. For example, QDA on proposed vector needs 0.07 seconds while it costs 0.1 seconds on natural vector. This is not a big difference in magnitude, however with a dataset which is much larger than ours, the classifiers on proposed vector would needs much less time. In addition, SVM is slow because of the long training period, reducing vectors from 12 dimension to 8 dimension in this case saves 65 s.

3.5.3 Virus family labels

The results are presented in Table 5. For proposed vectors on panel (a), prediction accuracy of KNN and SVM are greater than 0.85, and those of QDA, Logistic regression and Classification Tree are above 0.75. Meanwhile, all of those classifiers with accuracy greater than 0.75 have AUC close to 0.90, which means they are capable of correctly classifying the viruses at a very satisfactory probability level.

Based on Table 5 (b), we still observe slightly better performance metrics from classifiers on natural vector. Classifiers with highest accuracy are the same as in panel (a) only with the value of it being 0.89. Meanwhile, AUC values are 0.03 greater for KNN and QDA, and increased 0.01 for Logistic regression.

We have the similar observations on computing time that classifiers are generally work faster on proposed vectors than on natural vector except for Support Vector Machine. It takes 226.4 s for SVM on proposed vector whereas 203.4 s for SVM on natural vector. Although the difference is huge, costing more than 200 s is still way slower than those of other classifiers.

3.5.4 Protein family labels

Results are shown in Table 6. Classifiers on panel (a) except DLDA and Classification Tree have prediction accuracy above 0.80. Among them KNN is of best performance with highest AUC and Micro-F values. Those of LDA are very close to metrics of KNN. Logistic regression has satisfactory AUC and other metrics while SVM does not have relatively high AUC compared to others. Clearly, in this example, LDA works better than QDA. While surprisingly on panel (b), logistic regression has novel performance on natural vector given the AUC and micro-F metrics being 0.9663 and 0.9733 respectively. KNN and LDA also has relatively high performance.

Table 6: Results of protein families classification. The unit of time is a second (s).

Classifier	AUC	Micro-F	Time (s)
(a) Results of the proposed vector			
LDA	0.8541	0.8731	0.16 (0.02)
QDA	0.7287	0.8204	0.55 (0.04)
KNN	0.8986	0.8928	0.15 (0.01)
SVM	0.7895	0.8360	317.2 (0.6)
Logit	0.8465	0.8615	120.2 (1.2)
DLDA	0.8223	0.7317	0.09 (0.02)
Tree	0.7603	0.7827	2.02 (0.03)
(b) Results of Natural Vector			
LDA	0.9546	0.9687	0.27 (0.02)
QDA*	0.7144	0.7758	0.96 (0.23)
KNN	0.9144	0.9096	0.23 (0.01)
SVM	0.7650	0.7457	437.6 (0.7)
Logit	0.9663	0.9733	225.0 (1.0)
DLDA	0.8539	0.7995	0.13 (0.02)

Tree	0.7604	0.8407	1.91 (0.05)

On one hand, the computation time of logistic regression using the original natural vectors is 130 seconds worth of computation while LDA and KNN spend fewer than 0.30 seconds. On the other hand, KNN and LDA using the proposed vector run even faster, for their computation times are 0.23 seconds and 0.27 seconds, respectively. Even though the performances of LDA, QDA, and KNN using the proposed vector are a little worse than those using the original natural vectors, their accuracy rates and AUC are close to 0.9. Hence, the proposed vector is still considered satisfactory.

It is noteworthy that there is an $p > n$ problem (number of predictor is greater than number of observations) when implementing QDA on natural vector. The length of natural vector is 60 whereas the smallest protein family in our dataset only contains 52 members. It is obvious that covariance matrix for some families are singular, which makes it impossible for QDA to work. One of the remedy is to use regularized discriminant analysis (RDA). There are rich literature on this topic, for example, see (Friedman, 1989). Another choice is to use generalized inverse of covariance matrices in discriminant functions. Details on generalized inverse and its feasibility in discriminant analysis are discussed in (Rao & Mitra, 1972). The result for QDA in Table 6 (b), is obtained by using generalized inverse.

4 Discussion

Although the existing classification methods for predicting class labels of viruses or proteins have been successfully applied to various data, they may cost high computation time and in some cases they require high-performance computing platforms. Therefore, a simple solution with fast computing speed and satisfactory performance is always called for a dataset which consists of long genome sequences such as nucleotides or amino acids and an enormous number of resultant alignment-free vectors. Note that the central limit theorem can be applied to the modified natural vectors. Therefore, linear or quadratic discriminant analysis methods are applicable to the modified natural vectors, and those classifiers could be simply implemented and interpretable.

The real data analysis shows the proposed method's time consumption is very small and performance are satisfactory. Although the prediction accuracy of the proposed vector are slightly worse than the original natural vector in some cases, but the difference is not too much in terms of AUC and micro-F in general. Additionally, the modified natural vectors with LDA and QDA are faster than the original natural vectors. Therefore, the proposed method is convincing in terms of the minor performance and computing time trade-off. This phenomenon is due to the dimension reduction from natural vector to proposed vectors. For example, dimension is reduced by 4 for viral genome sequences and 20 for amino acids sequences.

Among the classifiers, LDA and QDA show top performances whereas KNN is also an satisfactory choice. However, there are still space for improvement, because the group size in our examples are not large, and then the asymptotic normality may not hold. Consequently, the Q-Q plots show some deviation from normal distribution for some families (Figure 1 and Figure 2). In addition, QDA has the singular covariance matrix problem if some of the class in training set is small. However, one can expect there is no such issue when dataset contains large number of sequences. Other classification methods such as logistic regression and SVM are not preferred since they are very time consuming. Moreover, DLDA and Tree classification methods have less classification accuracy and higher computing time than LDA and QDA, so that they are not recommended either (Table 4, Table 5, and Table 6).

The future work can be summarized into three directions. First, the proposed method could be applied to other sequences data, and also the normal approximation may be improved by enlarging sample sizes. Second, potential improvements in assessing the normality. Q-Q plots of the proposed vector indicates their distribution has heavy tails. One may consider do transformations on proposed vectors so that the distribution of resultant vectors are more close to normal distribution. Third, it exists the imbalanced classification problem in the protein families. Regularized discriminant analysis which balances between LDA and QDA may improve classification performance because it generally works for the situations of small sample sizes or when LDA and QDA are ill- or poorly-posed (Friedman, 1989).

References

Baltimore, D. (1971): "Expression of animal virus genomes," Bacteriol. Rev. 35 (3), 235–241.

Chan, R. H., R. W. Wang and H. M. Yeung (2010): "Composition vector method for phylogenetics-a review," Proc. 9th International Symposium on Operations Research and its Applications, 13–20.

Cortes, C. and V. Vapnik (1995): "Support-vector networks," Machine Learning, 20, 273–297.

Darling, D. A. (1975): "Note on a limit theorem," Ann. Probab. 3, 876–878.

Deng, M., C. Yu, Q. Liang, R. L. He, and S. S.-T. Yau (2011): "A Novel Method of Characterizing Genetic Sequences: Genome Space with Biological Distance and Applications," PLoS One, 6 (3), e17293.

Dudoit, S., J. Fridlyand, and T. P. Speed (2002): "Comparison of Discrimination Methods for the Classification of Tumors Using Gene Expression Data," J. Am. Stat. Assoc., 97, 77–87.

Fawcett, T. (2006): "An introduction to ROC analysis," Pattern Recognit. Lett., 27, 861–874.

Friedman, J. H. (1989): "Regularized discriminant analysis," J. Am. Stat. Assoc., 84, 165–175.

Ghor, B., D. Horn, N. Goldman, Y. Levy, and T. Massingham (2009): "Genomic DNA k-mer spectra: models and modalities," Genome Biol., 10, R108.

Hand, D. J. and R. J. Till (2001): "A simple generalisation of the area under the ROC curve for multiple class classification problems," Mach. Learn., 45: 171.

Hastie, T., R. Tibshirani, and J. Friedman (2009): The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition, Springer, New York.

Hernandez, T. and J. Yang (2013): "Descriptive statistics of the genome: phylogenetic classification of viruses," J. Comput. Biol., 23, 810–820.

Hoang, T., C. Yin, H. Zheng, C. Yu, L. R. He, and S. S.-T. Yau (2015): "A new method to cluster DNA sequences using Fourier power spectrum," J. Theor. Biol., 372, 135–145.

Huang, G. H., H. Q. Zhou, Y. F. Li, and L. X. Xu (2011): "Alignment-free comparison of genome sequences by a new numerical characterization," J. Theor. Biol., 281, 107–112.

Huang, G. H. (2014): "A novel neighborhood model to predict protein function from protein-protein interaction data," Current Bioinformatics," 11, 237–244.

Huang, H.-H., T. Xu, and J. Yang (2014a): "Comparing logistic regression, support vector machines, and permanental classification methods in predicting hypertension," BMC Proceedings, 8 (Suppl 1), S96.

Huang, H.-H., C. Yu, H. Zheng, T. Hernandez, S.-C. Yau, R. L. He, J. Yang, S. S.-T. Yau (2014b): "Global comparison of multiple-segmented viruses in 12-dimensional genome space," Mol. Phylogenet. Evol., 81, 29–36.

Huang, H.-H. (2016): "An ensemble distance measure of k-mer and Natural Vector for the phylogenetic analysis of multiple-segmented viruses," J. Theor. Biol., 398, 136–144.

Huang, G. H., C. Chu, T. Huang, X. Kong, Y. Zhang, N. Zhang, and Y.-D. Cai (2016): "Exploring mouse protein function via multiple approaches," PLoS One, 11, e0166580.

Huang, H.-H. and S.-B. Girimurugan (2018): "A novel real-time genome comparison method using discrete wavelet transform," J. Comput. Biol., 25, 405–416.

Maddouri, M. and M. Elloumi (2002): "A data mining approach based on machine learning techniques to classify biological sequences," Knowl. Based Syst., 15, 2002.

National Center for Biotechnology Information (NCBI)[Internet]. (2016): Bethesda (MD): National Library of Medicine (US), National Center for Biotechnology Information; Available from: https://www.ncbi.nlm.nih.gov/.

Polychronopoulos, D., E. Weitschek, S. Dimitrieva, P. Bucher, G. Felici, and Y. Almirantis (2014): "Classification of selectively constrained DNA elements using feature vectors and rule-based classifiers," Genomics 104, 79–86.

Rao, C. R. and S. K. Mitra (1972): "Generalized inverse of a matrix and its applications," Proc. Sixth Berkeley Symp. on Math. Statist. and Prob., Vol. 1, Univ. of Calif. Press, 601–620.

Selcuk, K., G. Dincer, and Z. Gokmen (2016): MVN: an R package for assessing multivariate normality. R package vignettes.

Sims, G. E., S. R. Jun, G. A. Wu, and S. H. Kim (2009): "Alignment-free genome comparison with feature frequency profiles (FFP) and optimal resolutions," Proc. Natl. Acad. Sci. U.S.A. 106, 2677–2682.

Vinga, S. and J. Almeida (2003): "Alignment-free sequence comparison review." Bioinformatics, 19, 513–523.

Vinga, S. (2007): Biological sequence analysis by vector-valued functions: revisiting alignment-free methodologies for DNA and protein classification. In: Pham, T. D., Yan, H., Crane, D. I. (Eds.), Advanced Computational Methods for Biocomputing and Bioimaging. Nova Science Publishers, New York.

Weitschek, E., F. Cunial and G. Felici (2015): "LAF: logic alignment free and its application to bacterial genomes classification," BioData Min., 8, 39.

Yu, C., T. Hernandez, H. Zheng, S.-C. Yau, H.-H. Huang, R. L. He, J. Yang, and S. S.-T. Yau (2013): "Real time classification of viruses in 12 dimensions," PLoS One, 8, e64328.