Computational and Structural Biotechnology Journal 23 (2024) 2083–2096

Contents lists available at ScienceDirect

Computational and Structural Biotechnology Journal

journal homepage: www.elsevier.com/locate/csbj

Research Article

The optimal metric for viral genome space

Hongyu Yu [a], Stephen S.-T. Yau [a,b,*]

[a] Department of Mathematical Sciences, Tsinghua University, Beijing, 100084, People's Republic of China
[b] Beijing Institute of Mathematical Sciences and Applications (Bimsa), Beijing, 101408, People's Republic of China

ARTICLE INFO

Dataset link: https://github.com/BobYHY/OptimalMetric

Keywords:
Alignment-free methods
Feature integration
Natural vector
Optimal metric
Viral genomes
Classification

ABSTRACT

Understanding the structural similarity between genomes is pivotal in classification and phylogenetic analysis. As the number of known genomes rockets, alignment-free methods have gained considerable attention. Among these methods, the natural vector method stands out as it represents sequences as vectors using statistical moments, enabling effective clustering based on families in biological taxonomy. However, determining an optimal metric that combines different elements in natural vectors remains challenging due to the absence of a rigorous theoretical framework for weighting different k-mers and orders. In this study, we address this challenge by transforming the determination of optimal weights into an optimization problem and resolving it through gradient-based techniques. Our experimental results underscore the substantial improvement in classification accuracy achieved by employing these optimal weights, reaching an impressive 92.73% on the testing set, surpassing other alignment-free methods. On one hand, our method offers an outstanding metric for virus classification, and on the other hand, it provides valuable insights into feature integration within alignment-free methods.

1. Introduction

The study of genome relationships has garnered significant attention in recent years as it provides a fundamental approach to understanding the connections among organisms. Traditional methods for sequence comparison rely on alignment, which, while effective, can be time-consuming [1–4]. With the advancement of sequencing techniques, the number of known genomes has increased rapidly. Consequently, more and more researchers are turning to alignment-free methods due to their high efficiency. A major idea in alignment-free methods involves embedding each genome to a point in the vector space. This transformation allows the sequence comparison problem to be recast as a classification or clustering problem for vectors, which can be readily solved using machine learning algorithms such as the K-NN method [5] or the K-means method [6]. In addition to its applications in classification and clustering, the concept of sequence embedding offers a novel approach to understanding genomes, representing each genome as a point in genome space and each family (or other levels of classification) as a cluster, providing a geometric perspective on genome analysis. In 2008, the Defense Advanced Research Projects Agency (DARPA) proposed two problems, namely "The Geometry of Genome Space" and "What are the Fundamental Laws of Biology?", along with 21 other challenges in pure and applied mathematics [7]. These challenges have spurred researchers from diverse academic backgrounds to investigate the genome space and its metrics.

There are various alignment-free methods for embedding sequences into vector spaces and defining metrics, mostly rooted in the analysis of k-mers from probabilistic, statistical, or information theory perspectives [8–13]. The natural vector (NV) method is an effective method that incorporates the concept of statistical moments, transforming sequences into feature information based on different k-mers and different moment orders [14,15]. Features here refer to various types of numerical elements extracted from the sequences. By assigning weights to various types of feature information, a comprehensive metric is established. Prior research has validated the effectiveness and efficiency of the NV-based method. Notably, the convex hulls formed by NVs from distinct families do not overlap, demonstrating that NVs belonging to the same family cluster together [16–18]. Furthermore, the comprehensive metric introduced by NV facilitates the efficient classification and phylogenetic analysis of biological sequences [15,17].

* Corresponding author at: Department of Mathematical Sciences, Tsinghua University, Beijing, 100084, People's Republic of China.
E-mail address: yau@uic.edu (S.S.-T. Yau).

https://doi.org/10.1016/j.csbj.2024.05.005
Received 28 November 2023; Received in revised form 22 April 2024; Accepted 4 May 2024
Available online 10 May 2024

Despite the success of NV-based methods, determining an optimal metric remains a challenging task due to the lack of a rigorous theoretical framework for weighing different k-mers and orders. In previous studies, these weights have been manually assigned [17]. While experimental results have shown the efficacy of such manual weight selection for real biological data, the search for an optimal weight for classification remains ongoing.

In this paper, we approach the weight selection as an optimization problem. We take a smooth approximation of the classification accuracy as the objective function and employ a modified version of the gradient descent method to calculate an optimal weight. The utilization of the optimal weight for classification yields an accuracy of 92.73% for the testing set, which is 4.88% higher than the best performance achieved by six other alignment-free approaches including both NV-based methods with manually determined weights and methods derived from other perspectives [17,11–13]. Moreover, we extend our analysis by applying the optimal weight to each Baltimore class and fine-tuning the weights within the classes. Subsequently, we construct phylogenetic trees based on the fine-tuned optimal weight. This research makes three significant contributions. Firstly, we present a rapid and accurate algorithm for classifying new genomes, which becomes increasingly vital as more genomes are discovered. Secondly, the distance metric derived from the optimal weight can be applied in the construction of phylogenetic trees for organisms, especially for viruses, where the absence of common genes found in cellular organisms poses a challenge [19]. Finally, our method offers an opportunity for the integration of the numerous alignment-free features currently available, facilitating further advancements in alignment-free methods.

2. Materials and methods

2.1. Dataset

The data utilized in this study comprise the complete virus reference sequences sourced from the National Center for Biotechnology Information (NCBI) up to June 30, 2022. The sequences can be accessed via the following URL: https://ftp.ncbi.nlm.nih.gov/refseq/release/viral. To ensure data quality, a data cleaning procedure was performed, which involved the removal of three types of sequences: (1) sequences containing nucleotides other than A, $T(U)$, C, G; (2) sequences lacking a family label; and (3) sequences belonging to families consisting of only a single sequence. Before the data cleaning process, there were 14,813 sequences. Following this process, the dataset retained a total of 11,559 sequences from 123 families. It is worth noting that our data contains multi-segment viruses, where each sequence represents one segment in this case. To establish the training and testing sets, 80% of the sequences were randomly selected as the training set, while the remaining sequences constituted the testing set. The Genbank IDs are listed in the Supplementary material and can also be found in https://github.com/BobYHY/OptimalMetric.

2.2. Natural vectors and k-mer natural vectors

The natural vector method is an alignment-free method that transforms DNA sequences into vectors of moments [14]. Consider the sequence $S = s_1 s_2 ... s_n$, define

$$w_k(s_i) = \begin{cases} 1, & s_i = k \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where $k, s_i \in \{A, T, C, G\}$. Then the natural vector of order m can be defined as

$$(n_A, n_C, n_G, n_T, \mu_A, \mu_C, \mu_G, \mu_T, D_2^A, D_2^C, D_2^G, D_2^T, ..., D_m^A, D_m^C, D_m^G, D_m^T) \tag{2}$$

where

$$\begin{cases} n_k = \sum_{i=1}^{n} w_k(s_i) \\ \mu_k = \sum_{i=1}^{n} \frac{i}{n_k} w_k(s_i) \\ D_j^k = \sum_{i=1}^{n} \frac{(i-\mu_k)^j}{n_k^{j-1} n^{j-1}} w_k(s_i) \\ n = n_A + n_T + n_C + n_G \end{cases} \tag{3}$$

n_k and μ_k are referred to as the order 0 element and order 1 element, respectively. D_j^k denotes the order j element.

The k-mer natural vector method is an extension of the natural vector method [15]. K-mer is a string composed of k nucleotides and there are 4^k possible k-mers (denoted by $l_1, ..., l_{4^k}$). For the sequence $S = s_1 s_2 ... s_n$, we can regard it as a sequence consisting of $n - k + 1$ k-mers $(s_1 ... s_k)...(s_{n-k+1} ... s_n)$. Similar to traditional natural vectors, we can define the k-mer natural vector

$$(n_{l_1}, ..., n_{l_k}, \mu_{l_1}, ..., \mu_{l_k}, D_2^{l_1}, ..., D_2^{l_k}, ..., D_m^{l_1}, ..., D_m^{l_k}).$$

(If $n_{l_i} = 0$, we let $\mu_{l_i} = D_2^{l_i} = ... = 0$.)

2.3. The optimal weight and the algorithm for training

There are elements of different k-mers (1-K) and different orders (0-M), as mentioned before. Let $dis_{km}(i, j)$ represent the Euclidean distance between the k-mer order m elements of sequence i and sequence j. By assigning a weight w_{km} ($k = 1, ..., K$; $m = 0, ..., M$) to each distance, we can formulate a weighting metric:

$$Dis^w(i, j) = \sum_{k=1}^{K} \sum_{m=0}^{M} w_{km} dis_{km}(i, j). \tag{4}$$

To determine the optimal weight for classification purposes, we need a scoring criterion to evaluate different weights. In other words, we should develop a smooth function that quantifies the effectiveness of a given weight w. In the case of sequence classification using natural vectors, the 1-nearest neighbor (1-NN) method with the leave-one-out strategy is commonly employed [5]. Therefore, a natural approach is to utilize the accuracy of predictions obtained from the 1-NN method with the leave-one-out strategy as the score for a particular weight, i.e.,

$$S(w) := \frac{1}{N}\sum_{i=1}^{N} 1_{\{F(i)=F(\arg\min_{j\neq i}\{Dis^w(i,j)\})\}} \tag{5}$$

where N is the number of the sequences, 1_A is the indicator function of the set A and $F(i)$ is the family that the i-th sequence belongs to.

However, $S(w)$ defined above is not continuous for w so it is very complicated to optimize. Therefore, we consider a smooth approximation of $S(w)$. Let $f_n(x) = \frac{1}{x^n}$, we define

$$S_n(w) := \frac{1}{N}\sum_{i=1}^{N} C_i(w) \tag{6}$$

where

$$C_i(w) := \frac{\sum_{F(j)=F(i),j\neq i} f_n(Dis^w(i,j))}{\sum_{j\neq i} f_n(Dis^w(i,j))}. \tag{7}$$

Given a fixed weight $w^{(0)}$ and a fixed integer i_0, suppose $j_0 = \arg\min_{j\neq i_0}\{Dis^{w^{(0)}}(i_0,j)\}$ is well-defined (the nearest neighbor is unique), then we can prove that $\lim_{n\to+\infty} C_{i_0}(w^{(0)}) = 1_{\{F(i_0)=F(j_0)\}}$ and therefore $\lim_{n\to+\infty} S_n(w^{(0)}) = S(w^{(0)})$. The proof is simple:

$$Dis^{w^{(0)}}(i_0,j_0) < \min_{j\neq j_0, j\neq i_0} Dis^{w^{(0)}}(i_0,j)$$

$$f_n(Dis^{w^{(0)}}(i_0,j_0)) > \max_{j\neq j_0, j\neq i_0} f_n(Dis^{w^{(0)}}(i_0,j))$$

$$\lim_{n\to+\infty} \frac{\max_{j\neq j_0, j\neq i_0} f_n(Dis^{w^{(0)}}(i_0,j))}{f_n(Dis^{w^{(0)}}(i_0,j_0))} = 0$$

Therefore, when n is sufficiently large, the element $f_n(Dis^{w^{(0)}}(i_0,j_0))$ becomes much larger than any other elements in the fraction of C_i, rendering the other elements negligible. That is, if $F(i_0) = F(j_0)$, then $\lim_{n\to+\infty} C_{i_0}(w^{(0)}) = 1$; otherwise, $\lim_{n\to+\infty} C_{i_0}(w^{(0)}) = 0$.

Therefore, we can approximate S by S_n and our goal is to solve the following optimization problem given n_0, K, and M:

$$\begin{aligned} &\max \quad S_{n_0}(w) \\ &\text{s.t.} \quad w_{km} \geq 0, \quad k = 1,...,K;\ m = 0,...,M. \end{aligned} \tag{8}$$

The gradient descent method is a traditional optimization algorithm for unconstrained problems. It can also be applied in constrained cases by mapping the result in each step to the feasible region. Let $ReLU(x) = \max(x,0)$ and this calculation can be broadcast to vectors, then the optimization problem (8) can be solved by the iteration below:

$$w^{(n+1)} = ReLU(w^{(n)} + l\nabla S_{n_0}(w^{(n)})) \tag{9}$$

where l is the learning rate.

Many modified versions of the gradient descent method such as stochastic modifications like Adam [20] have been proposed. However, we found through numerical experiments that utilizing these stochastic modifications in Algorithm 1 did not yield favorable results. Therefore, they were not implemented. We adopt the idea of the backtracking line search to this problem and propose the following algorithm.

Algorithm 1 The algorithm that solves the optimization problem (8).

Require: Parameters n_0, K, M, the learning rate l, patience P, and weight magnitude A_{km}.
Ensure: Trained weight w.

```
Let i = 0
Randomly generate the weight w^(0) = (w_km^(0)) (w_km^(0) ~ Uniform[0, A_km]).
while i <= P do
    v = l||w^(i)||_{L1} ∇S_{n0}(w^(i)) / ||∇S_{n0}(w^(i))||_{L1}
    while S_{n0}(ReLU(w^(i) + v)) < S_{n0}(w^(i)) do
        v = (1/2) v
    end while
    w^(i+1) = ReLU(w^(i) + v)
    if S_{n0}(w^(i+1)) == S_{n0}(w^(i)) then
        i = i + 1
    else
        i = 0
    end if
end while
```

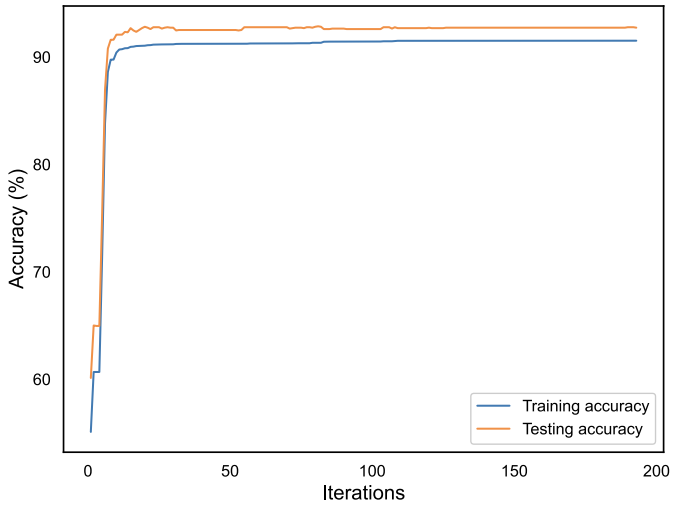

Fig. 1. The progression of training and testing accuracy for classification during the training process.

In this paper, we choose $n_0 = 45$, $K = 9$, $M = 2$, $l = 0.1$, and $P = 50$. We choose n_0 to be sufficiently large, but not excessively so, to prevent exceeding the numerical bounds when computing S_{n_0}. The choice of K and M is based on previous research [17]. The learning rate l represents the proportion of updates to the original weights. We select it to be a substantial yet not excessive quantity. For P, we opt for a value that is sufficiently large.

In addition, we choose A_{km} such that the mean of elements in the initial weight is inversely proportional to the mean of the corresponding distance matrix, i.e., $A_{km}E[dis_{km}] = constant$, which avoids information to be ignored due to magnitude.

We implement the algorithm in the pytorch framework [21]. The code can be found in both the Supplementary material and the Github repository https://github.com/BobYHY/OptimalMetric.

2.4. The phylogenetic analysis

To construct a phylogenetic tree, we utilize the Hausdorff distance to measure the distance between families. The Hausdorff distance quantifies the extent of separation between two subsets. Its definition is as follows:

$$d_H(X,Y) = \max\left\{\sup_{x\in X} d(x,Y), \sup_{y\in Y} d(X,y)\right\}. \tag{10}$$

Hausdorff distance has been found to perform well between sets of natural vectors in previous studies, and it follows the triangular inequality, which makes it a well-defined distance in mathematics [22]. Using the distance matrix obtained from the Hausdorff distance, we employ the BioNJ algorithm [23], which is an enhanced version of the neighbor-joining algorithm [24], to construct the tree. Our algorithm is implemented online through the following website: http://www.atgc-montpellier.fr/fastme/ [25]. The trees are visualized using iTOL [26].

3. Results

3.1. The classification performance

Our objective is to effectively classify the viruses into their respective families by determining an optimal metric. To achieve this, we calculate the optimal weight for each k-mer and each moment order based on the training set. Subsequently, we apply the metric, which is induced by the optimal weight, to the testing set.

The optimization of the weight is achieved through the smooth approximation of the accuracy of the classification. During the training process, consisting of 193 iterations, we observe the effectiveness of the approximation. The maximum difference between the training accuracy and its approximation during the training process is found to be only 0.005%. This result eliminates the necessity to differentiate between these two concepts in subsequent discussions.

The progression of training and testing accuracy during the training process is depicted in Fig. 1. Notably, the training accuracy exhibits a remarkable increase, starting from 55.10% and reaching 91.52%. Similarly, the testing accuracy also shows a significant improvement, rising from 60.12% to 92.73%. The rapid and substantial growth in accuracy is evident. The simultaneous increase of both indicates that the knowledge gained about weight selection from the training set is generalizable rather than a result of overfitting. It is noteworthy that the testing accuracy outperforms the training accuracy due to the implementation of a leave-one-out strategy in this paper. This strategy entails predicting the outcomes for the testing set using all other sequences in both the training and testing sets, mimicking real-world scenarios. Conversely, predictions for the training set solely rely on other sequences within the training set to prevent data leakage.

We compare this classification accuracy with other methods. To begin with, we compare it with methods based on NV, but with manually determined weights. These methods do not differentiate weights based on different orders but only distinguish between weights for different k-mers. In the notation used in this paper, we can view the corresponding metric as

$$Dis^a = \sum_{k=1}^{K} a_k \sqrt{\sum_{m=0}^{2} dis_{km}^2}. \tag{11}$$

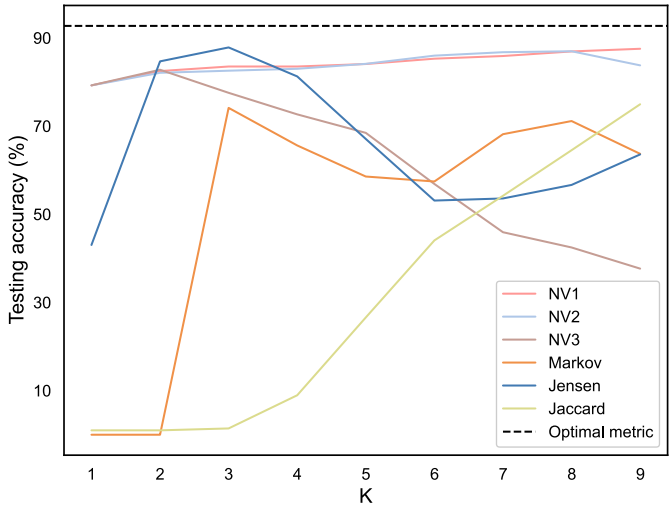

Fig. 2. Comparison of classification accuracy between our method and six other alignment-free methods.

Table 1
The number of sequences from each Baltimore class.

Baltimore class	The training set	The testing set
I	4075	1040
II	1161	308
III	1114	282
IV	1452	349
V	1288	290
VI	68	17
VII	89	26

In previous studies, three types of manually determined weights are commonly used: $a_k = \frac{1}{2^k}$, $a_k = \frac{1}{k^2}$, and $a_k = 1_{\{k=K\}}$. We refer to methods using these three weights as NV1, NV2, and NV3, respectively. We evaluated and examined the accuracy of these three methods on the testing set, varying K from 1 to 9. The best result was achieved by the 9-mer NV1 method, with an accuracy of 87.54%. (See Fig. 2.) Notably, the optimal metric significantly outperforms manually determined weights.

Then, we compare our method with three other alignment-free algorithms based on different theories. The first method, denoted as the Markov method, originates from [11]. It is based on a Markov model and corrects random background information present in K-mers using information from $(K-1)$-mers and $(K-2)$-mers $(K>2)$ from the perspective of probability. The second method, denoted as the Jensen method, is from [12] and uses Jensen-Shannon divergence, a concept in information theory, to measure the distance between feature frequency profiles. The third method, denoted as the Jaccard method [13], calculates distances based solely on the presence of features using the Jaccard distance. Similarly, we conducted an evaluation by varying K from 1 to 9 and examined the accuracy of these three methods on the testing set. The results show that the best accuracy was achieved by the 3-mer Jensen method, with an accuracy of 87.85%. (See Fig. 2.) Our method significantly outperforms these methods by 4.88%.

3.2. Fine-tuning within the Baltimore classes

Previously, we have verified the reliability of the optimal metric for viral genomes. For specific task scenarios, we can further fine-tune the weights of this metric according to requirements to enhance its performance. For example, if we already know the Baltimore class to which a virus belongs and need to perform classification within that class, we would need to fine-tune the metric using the dataset specific to that class. The fine-tuning process utilizes the same training method employed to obtain the optimal weight for the entire dataset. However, in this case, the training set is restricted to a subset of the complete training set, and the process begins from the previously obtained optimal weight.

In this paper, we focus on fine-tuning the optimal weight for 7 Baltimore classes. The Baltimore classification system categorizes viruses into 7 classes based on the type of genome molecule and replication strategy [27–29]. These classes include double-stranded DNA viruses, single-stranded DNA viruses, double-stranded RNA viruses, positive-sense single-stranded RNA viruses, negative-sense single-stranded RNA viruses, single-stranded RNA reverse transcriptase viruses, and double-stranded DNA reverse transcriptase viruses. To construct new training and testing sets, we extract the sequences belonging to each specific Baltimore class from the original training and testing sets. Table 1 provides the corresponding sequence numbers for these subsets.

Table 2 presents the testing accuracy for each Baltimore class using both the initial optimal weight and the weight after fine-tuning for each class. For Baltimore class VI and Baltimore class VII, the testing accuracy before fine-tuning is already 100%, indicating that retraining is unnecessary. In the case of Baltimore class I, II, III, and V, the testing accuracy before fine-tuning is satisfactory, but retraining can still lead to improvements. However, for Baltimore class IV, fine-tuning may result in over-fitting, where the testing accuracy decreases as the training accuracy increases. In summary, our findings indicate that the classification performance in subsets is generally good even before fine-tuning. Moreover, we have observed that fine-tuning can further enhance the performance in many cases.

Table 2
The testing accuracy for each Baltimore class before and after fine-tuning.

Baltimore class	Before fine-tuning	After fine-tuning
I	94.03%	95.19%
II	96.90%	97.72%
III	97.12%	97.51%
IV	92.07%	90.83%
V	88.98%	90.00%
VI	100%	-
VII	100%	-

Fig. 3. The optimal weight for each order and k-mer.

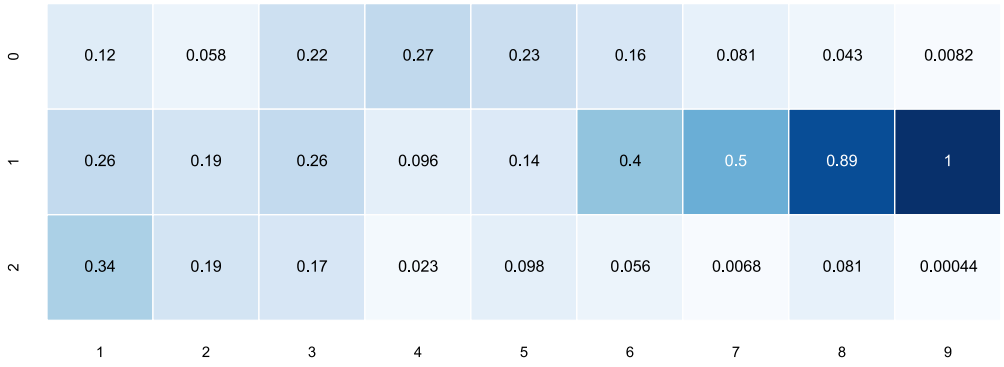

Fig. 4. The importance corresponding to the optimal weight.

3.3. The optimal weight and the corresponding importance

Now we turn our attention to the optimal weights themselves. The weight w_{km} considered in this study is a 27-dimensional vector assigned to the statistical moments for 1-9-mers and orders 0-2. Fig. 3 presents a visualization of the optimal weight before fine-tuning. Normalization is performed to ensure that its maximum value is 1. Each row represents the weight for a specific order, and each column represents the weight for a particular k-mer. The visualization reveals that the weights for order 1 and order 2 elements tend to 0 as k increases, while the weight for order 0 elements initially increases and then decreases with the growth of k. This optimal weight provides valuable insights into integrating various statistical information within a biological sequence.

However, it is important to note that a high weight assigned to an element does not necessarily indicate its significant role in the classification. For instance, if two elements have similar weights but their corresponding moments have significantly different magnitudes, their importance in the classification task will not be equal. To address this, we introduce the concept of element importance. Let w_{kj} denote the weight for k-mer and order j, and $E[dis_{kj}]$ represent the mean distance for k-mer and order j. The importance I_{kj} of each element is defined as the product of its weight and mean distance, i.e., $I_{kj} = w_{kj} \times E[dis_{kj}]$. We utilize the concept of importance to further investigate the significance of each element. Fig. 4 illustrates that the order 1 elements with large values of k, particularly 6-9-mers, hold great importance. This observation provides an explanation for the notable improvement in classification results with the inclusion of higher k-mers in previous studies [17].

We also provide visualizations of the weight and the corresponding importance after fine-tuning in Fig. A.7–A.14. We can observe that the weight undergoes only slight changes, whereas the corresponding importance exhibits more significant variations. Furthermore, we find that the previous observations hold true after fine-tuning. The weight pattern remains consistent with what is shown in Fig. 3, with elements having high k values in order 1 continuing to exhibit significant importance. The stability of these features indicates that the optimal metric is not merely a result specific to a particular dataset but possesses a degree of generality.

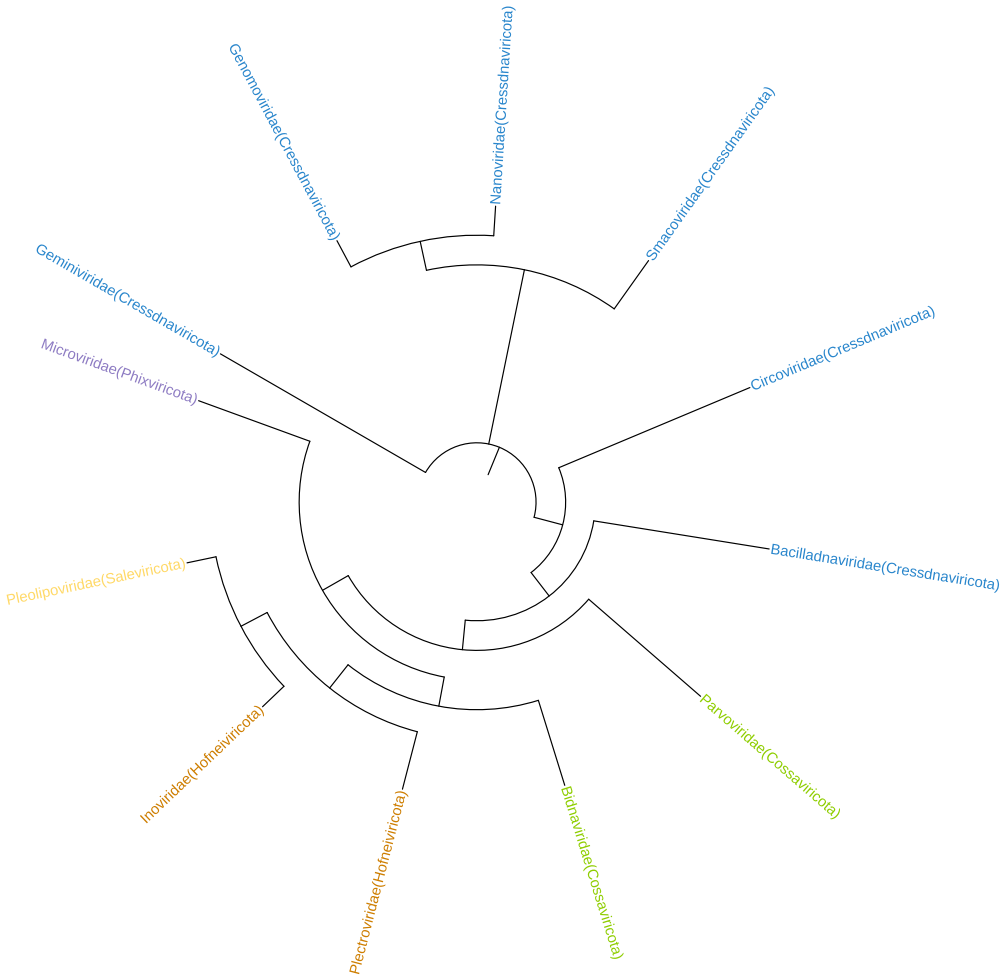

Fig. 5. The phylogenetic tree for Baltimore class II based on the optimal weight after fine-tuning.

3.4. The phylogenetic analysis for each Baltimore class

The optimal metric we have obtained provides a novel approach to constructing phylogenetic trees. Instead of relying on the identification and alignment of a common gene, which is challenging to find for viruses, we can directly utilize the entire genomes and use statistical moments to define the distance. Once a suitable metric for the genome is determined, we can extend it to define a metric for virus families using the Hausdorff distance. This metric enables us to perform phylogenetic analysis. In our study, we employ the BioNJ method [23] to construct phylogenetic trees for each Baltimore class.

Fig. 5 illustrates the phylogenetic tree for Baltimore class II, generated using the optimal weight after fine-tuning. Additional phylogenetic trees for other Baltimore classes are shown in Fig. A.15–A.18. It is worth noting that for class VI and class VII, there are insufficient families to construct trees. As for class IV, the weight before fine-tuning is applied due to the over-fitting issue.

We compared our phylogenetic results with those of the previous study [30]. At the phylum level, the previous study emphasized the similarities between Cossaviricota and Cressdnaviricota, which aligns with our findings. At the family level, most families within the same phylum exhibit close relationships in this tree. However, there are a few exceptions; for instance, two families within Hofneiviricota did not cluster together in a single branch. Similar phenomena are observed in other Baltimore classes. Overall, while our method can provide phylogenetic results of reference value, it does not ensure a comprehensive depiction of relationships between families. This limitation may stem from our optimal weight being trained on predicting family labels, which might not fully capture the relationship between families.

3.5. The impact of different taxonomic standards

Our classification of sequences is based on taxonomic standards determined manually. Different standards yield different family classifications for sequences, thus affecting the classification accuracy. In our previous analysis, we utilized taxonomic standards as of June 30, 2022, to maintain time consistency between the data and its annotations. To further illustrate the effectiveness of our method, we validate its performance under alternative taxonomic standards. We update the annotations to adhere to standards as of March 7, 2024 and employ the same data cleaning process. This results in 11,422 sequences from 148 families, which can also be found in https://github.com/BobYHY/OptimalMetric. (The variation in the number of families results from the splitting or restructuring of certain previous families into smaller units, while the alteration in the number of sequences arises from the cleaning process.)

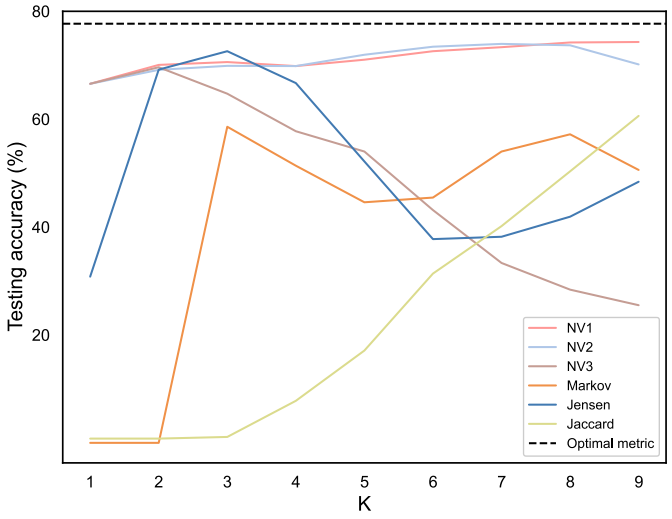

Fig. 6. Comparison of classification accuracy in the latest taxonomic standard.

We repeat testing accuracy calculations in section 3.1. The testing accuracy achieved using the optimal metric was 77.7%. In comparison, the highest accuracy achieved by the NV1, NV2, NV3, Markov, Jensen, and Jaccard methods under different K-values were 74.3%, 74.0%, 69.6%, 58.6%, 72.6%, and 60.6% respectively. (See Fig. 6.)

We can derive two insights from this comparison. First, our method remains superior to others under the new taxonomic standards, regardless of whether based on natural vectors or other approaches. This underscores both the effectiveness of our method and its adaptability to different standards. Second, the comparison between different taxonomic standards reveals a lower accuracy under the new standards compared to the previous ones. This could be attributed to several factors: Firstly, the increased number of families under the new taxonomic standards makes classification more challenging. Secondly, the old taxonomic standards align with the time of data acquisition, suggesting the need to incorporate all the latest data for improved classification under the new standards. Lastly, there may be issues with the setup of the new taxonomic standards that require improvement.

4. Discussion

In this paper, we started with the idea of maximizing classification capability and utilized the weight training approach to obtain the optimal alignment-free algorithm based on statistical moments. Then, we analyzed all viral reference sequences and calculated the optimal metric for viral genome space as Formula (12). (Please refer to Fig. 3 for the specific weights.) We validated its excellent classification performance.

$$\begin{aligned} Dis^{w} &= \sum_{k=1}^{9}\sum_{m=0}^{2} w_{km} dis_{km} \\ &= 3.5\times 10^{-2} dis_{1,0} + \ldots + 5.1\times 10^{-6} dis_{9,2}. \end{aligned} \tag{12}$$

In future applications of this method, there is no longer a need to retrain the optimal metric. Instead, the weights calculated above can be directly applied. For instance, when given an unknown viral sequence, we can utilize these weights to calculate its distance from other sequences in the database directly, thus determining its most likely family membership.

Our method has two major advantages compared to mainstream alignment algorithms. Firstly, being alignment-free, this method provides a substantial increase in speed. For a sequence set of length N, where each sequence has a length of $O(L)$, the time complexity of performing multiple sequence alignment (MSA) is $O(L^N)$, while the time complexity of performing pairwise alignments for all sequences is $O(N^2L^2)$. However, with our method, computing k-mer natural vectors and generating the distance matrix has a time complexity of $O(NL + N^2 4^k)$. Even when $k = 9$, this complexity is significantly lower than that of using alignment methods on viral datasets. (In our dataset, the average sequence length is 34,872.) Secondly, in sequence comparison, we do not depend on conserved segments. Instead, we compare statistical patterns from a higher perspective. This allows us to offer high-quality analysis for sequences like viral sequences that do not contain conserved regions.

In comparison to other alignment-free methods, our algorithm also exhibits three major advantages. Firstly, previous alignment-free methods extracted valuable features, but the systematic integration of these features has not been thoroughly studied. Our algorithm provides a method for integrating information from an optimization perspective, offering new insights for subsequent research on alignment-free methods. Secondly, our method can self-adjust weights based on different datasets, reducing the impact of sequence type differences to some extent. Thirdly, experimental results demonstrate that our algorithm's classification performance is significantly superior to previous algorithms.

Finally, from a geometric perspective, the optimal metric itself holds intrinsic significance. Past research on the geometry of genome space based on NV methods has extracted a series of geometric principles including the convex hull principle. However, the study of the metric itself has remained limited to empirical approaches. The metric serves as the foundation for the geometric structure. The optimal metric we have extracted sheds light on the manifold structure of the genome space to some extent.

Certainly, our method currently still has limitations that require further investigation in future studies. First, our objective function used for training is non-convex, which means that the uniqueness of the optimal solution and the global optimization cannot be guaranteed. While the practical significance of these locally optimal weights has been demonstrated, a more unique determination of the optimal solution in subsequent studies would further enhance the geometric interpretation of this optimal metric. Second, our method is only suitable for relatively complete

sequences. If the data collected consists of only small fragments, such as in metagenomics, further research is needed to determine how to use our method to identify the types of these fragments.

Funding

This research was funded by National Natural Science Foundation of China (NSFC) grant (12171275) and Tsinghua University Education Foundation fund (042202008).

CRediT authorship contribution statement

Hongyu Yu: Conceptualization, Data curation, Formal analysis, Methodology, Software, Validation, Visualization, Writing – original draft. **Stephen S.-T. Yau:** Conceptualization, Funding acquisition, Methodology, Project administration, Resources, Supervision, Writing – review & editing.

Declaration of competing interest

None.

Data availability

The data and code that support the findings of this study are available in https://github.com/BobYHY/OptimalMetric.

Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work the authors used ChatGPT in order to improve language. After using this tool, the authors reviewed and edited the content as needed and takes full responsibility for the content of the publication.

Appendix A

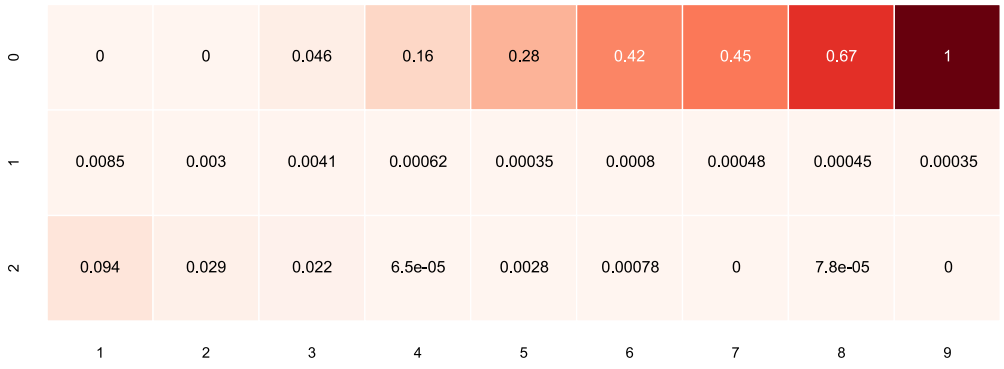

Fig. A.7. The optimal weight after fine-tuning (Baltimore class I).

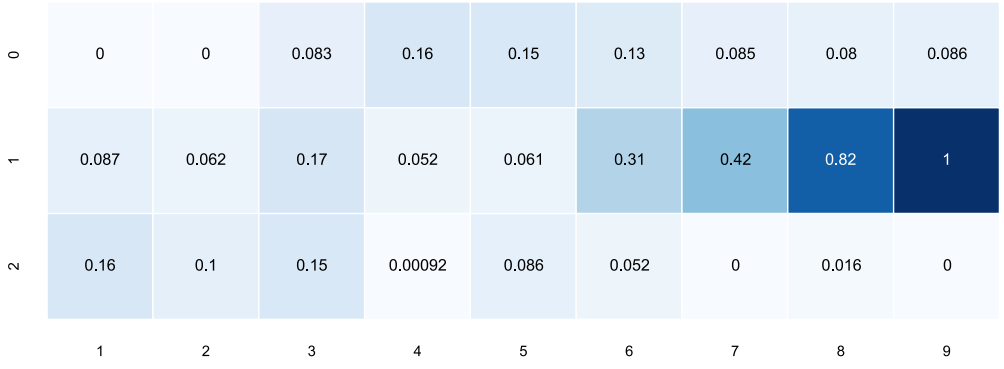

Fig. A.8. The importance after fine-tuning (Baltimore class I).

Fig. A.9. The optimal weight after fine-tuning (Baltimore class II).

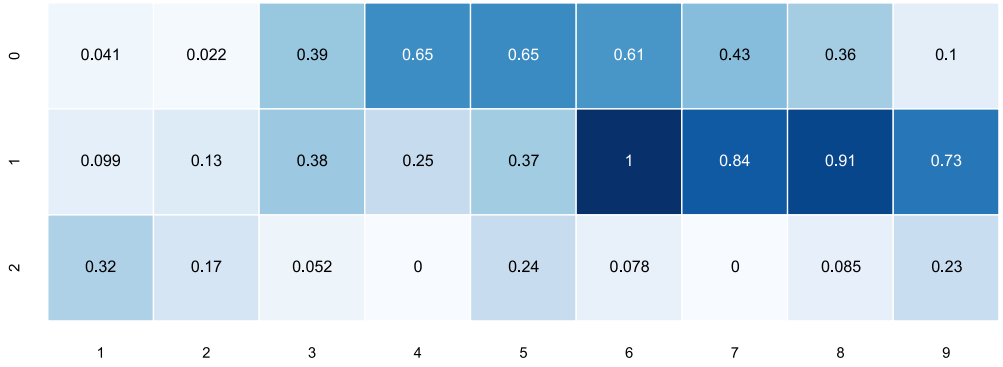

Fig. A.10. The importance after fine-tuning (Baltimore class II).

Fig. A.11. The optimal weight after fine-tuning (Baltimore class III).

Fig. A.12. The importance after fine-tuning (Baltimore class III).

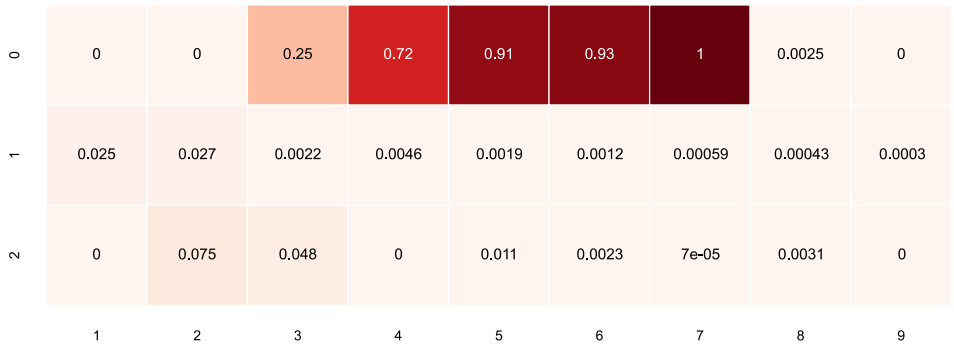

Fig. A.13. The optimal weight after fine-tuning (Baltimore class V).

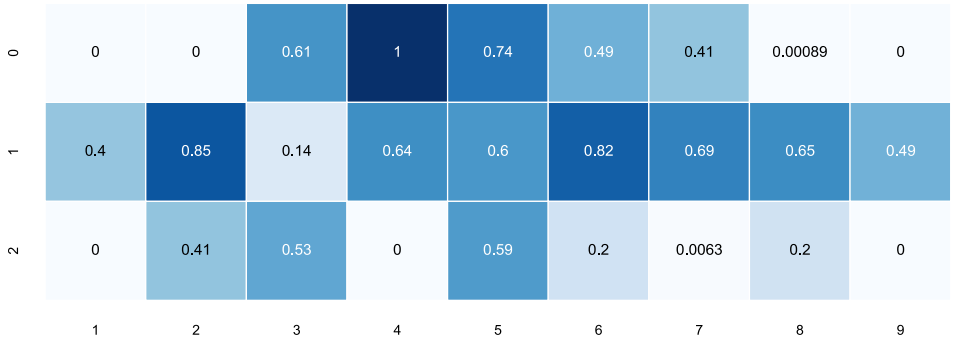

Fig. A.14. The importance after fine-tuning (Baltimore class V).

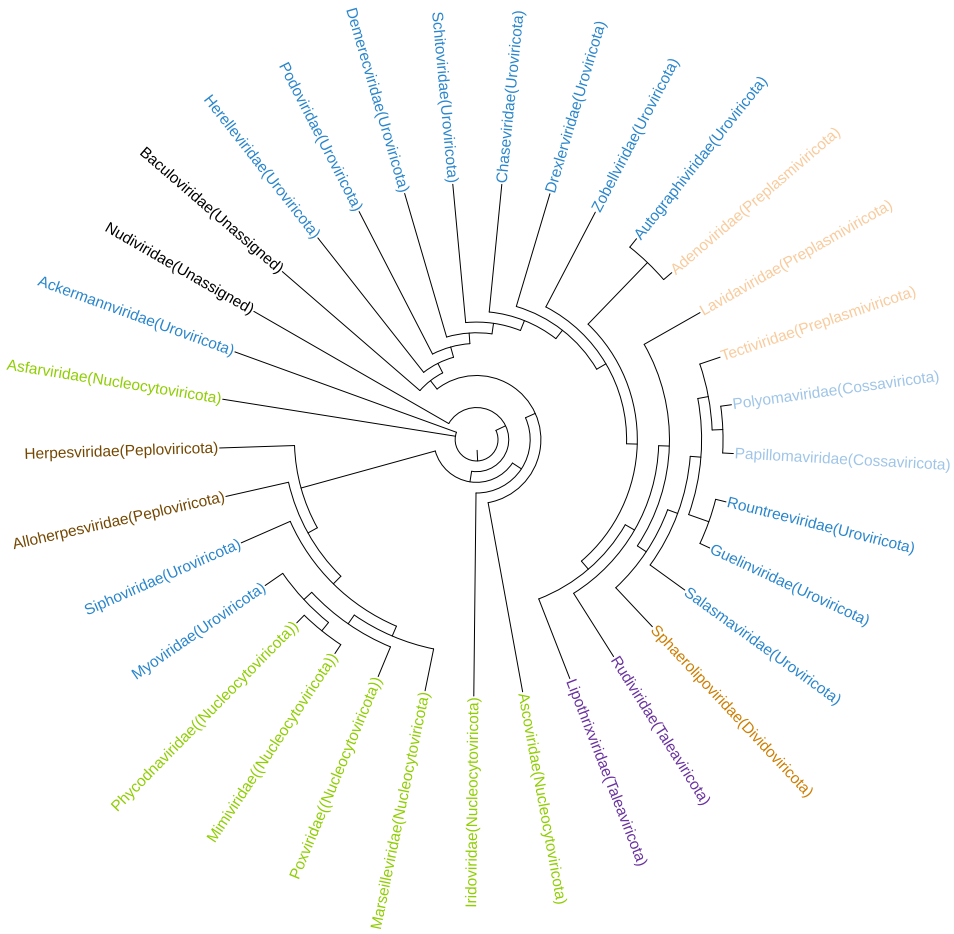

Fig. A.15. The phylogenetic tree for Baltimore class I based on the optimal weight after fine-tuning.

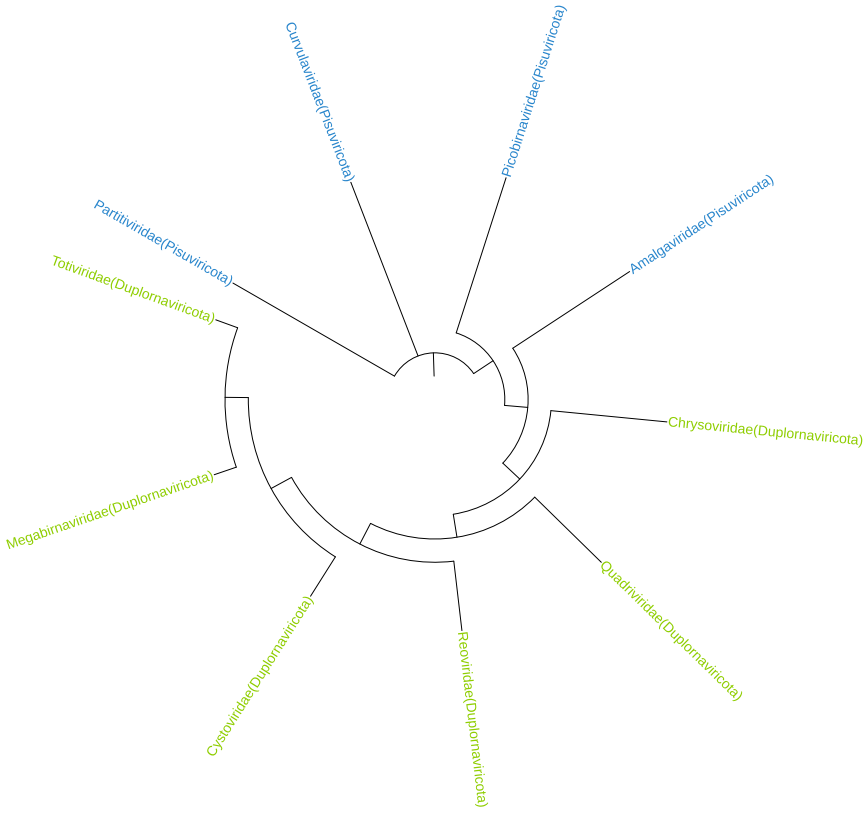

Fig. A.16. The phylogenetic tree for Baltimore class III based on the optimal weight after fine-tuning.

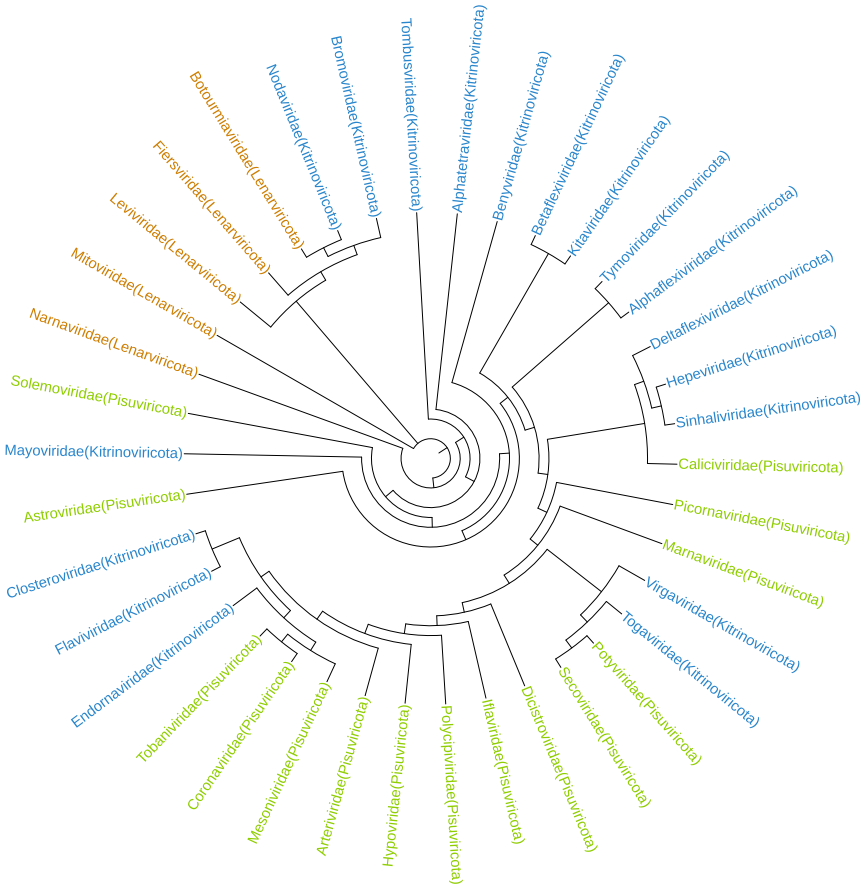

Fig. A.17. The phylogenetic tree for Baltimore class IV based on the optimal weight before fine-tuning.

Fig. A.18. The phylogenetic tree for Baltimore class V based on the optimal weight after fine-tuning.

Appendix B. Supplementary material

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.csbj.2024.05.005.

References

[1] Needleman SB, Wunsch CD. A general method applicable to the search for similarities in the amino acid sequence of two proteins. J Mol Biol 1970;48:443–53.
[2] Smith TF, Waterman MS. Identification of common molecular subsequences. J Mol Biol 1981;147:195–7.
[3] Edgar RC. Muscle: multiple sequence alignment with high accuracy and high throughput. Nucleic Acids Res 2004;32:1792–7.
[4] Higgins DG, Sharp PM. CLUSTAL: a package for performing multiple sequence alignment on a microcomputer. Gene 1988;73:237–44.
[5] Cover TM, Hart PE. Nearest neighbor pattern classification. IEEE Trans Inf Theory 1967;13:21–7.
[6] Hartigan JA, Wong MA. A k-means clustering algorithm; 1979.
[7] DARPA. Broad agency announcement (BAA 07-68) for Defense Sciences Office (DSO). http://www.math.utk.edu/~vasili/refs/darpa07.MathChallenges.html, 2008.
[8] Zielezinski A, Vinga S, Almeida JS, Karłowski WM. Alignment-free sequence comparison: benefits, applications, and tools. Genome Biol 2017;18.
[9] Bonham-Carter O, Steele J, Bastola DR. Alignment-free genetic sequence comparisons: a review of recent approaches by word analysis. Brief Bioinform 2014;15:890–905.
[10] Lu YY, Tang K, Ren J, Fuhrman JA, Waterman MS, Sun F. CAFE: aCcelerated Alignment-FrEe sequence analysis. Nucleic Acids Res 2017;45:W554–9.
[11] Qi J, Wang B, Hao B. Whole proteome prokaryote phylogeny without sequence alignment: a k-string composition approach. J Mol Evol 2003;58:1–11.
[12] Jun SR, Sims GE, Wu GA, Kim SH. Whole-proteome phylogeny of prokaryotes by feature frequency profiles: an alignment-free method with optimal feature resolution. Proc Natl Acad Sci USA 2010;107:133–8.
[13] Levandowsky M, Winter DK. Distance between sets. Nature 1971;234:34–5.
[14] Deng M, Yu C, Liang Q, He RL, Yau SST. A novel method of characterizing genetic sequences: genome space with biological distance and applications. PLoS ONE 2011;6:e17293.
[15] Wen J, Chan RH, Yau SC, He RL, Yau SST. K-mer natural vector and its application to the phylogenetic analysis of genetic sequences. Gene 2014;546:25–34.
[16] Zhao X, Tian K, He RL, Yau SST. Convex hull principle for classification and phylogeny of eukaryotic proteins. Genomics 2018;111:1777–84.
[17] Sun N, Pei S, He L, Yin C, He RL, Yau SST. Geometric construction of viral genome space and its applications. Comput Struct Biotechnol J 2021;19:4226–34.
[18] Tian K, Zhao X, Yau SST. Convex hull analysis of evolutionary and phylogenetic relationships between biological groups. J Theor Biol 2018;456:34–40.
[19] Harris HMB, Hill C. A place for viruses on the tree of life. Front Microbiol 2021;11.
[20] Kingma D, Ba J. Adam: a method for stochastic optimization. In: International conference on learning representations; 2014.
[21] Paszke A, Gross S, Massa F, Lerer A, Bradbury J, et al. PyTorch: an imperative style, high-performance deep learning library. Adv Neural Inf Process Syst 2019;32.
[22] Huang HH, Yu C, Zheng H, Hernandez T, Yau SST, He R, et al. Global comparison of multiple-segmented viruses in 12-dimensional genome space. Mol Phylogenet Evol 2014;81.
[23] Gascuel O. BIONJ: an improved version of the NJ algorithm based on a simple model of sequence data. Mol Biol Evol 1997;14:685–95.
[24] Saitou N, Nei M. The neighbor-joining method: a new method for reconstructing phylogenetic trees. Mol Biol Evol 1987;4:406–25.
[25] Lefort V, Desper R, Gascuel O. FastME 2.0: a comprehensive, accurate, and fast distance-based phylogeny inference program. Mol Biol Evol 2015;32:2798–800.
[26] Letunić I, Bork P. Interactive tree of life (iTOL) v5: an online tool for phylogenetic tree display and annotation. Nucleic Acids Res 2021;49:W293–6.

[27] Baltimore D. Expression of animal virus genomes. Bacteriol Rev 1971;35:235–41.
[28] Baltimore D. Viral genetic systems. Trans N Y Acad Sci 1971;33:327–32.
[29] Baltimore D. The strategy of RNA viruses. Harvey Lect 1974;70:57–74.
[30] Koonin E, Kuhn J, Dolja V, Krupovic M. Megataxonomy and global ecology of the virosphere. ISME J 2024;18.