Gene 962 (2025) 149532

Contents lists available at ScienceDirect

Gene

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

Research paper

Novel natural vector with asymmetric covariance for classifying biological sequences

Guoqing Hu [a,*,1], Tao Zhou [b,1], Piyu Zhou [a,c,d], Stephen Shing-Toung Yau [a,b,*]

[a] Beijing Institute of Mathematical Sciences and Applications (BIMSA), 101408, Beijing, China
[b] Department of Mathematical Sciences, Tsinghua University, 100084, Beijing, China
[c] State Key Laboratory of Mathematical Science, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, 100190, Beijing, China
[d] University of Chinese Academy of Sciences, 100049, Beijing, China

ARTICLE INFO

Edited by Eric Kmiec

Keywords:
Asymmetric covariance
Sequence comparison
Natural vector
Convex hull principle

ABSTRACT

The genome sequences of organisms form a large and complex landscape, presenting a significant challenge in bioinformatics: how to utilize mathematical tools to describe and analyze this space effectively. The ability to compare relationships between different organisms depends on creating a rational mapping rule that can uniformly encode genome sequences of varying lengths as vectors in a measurable space. This mapping would enable researchers to apply modern mathematical and machine learning techniques to otherwise challenging genomic comparisons. The natural vector method has been proposed as a concise and effective approach to accomplish this. However, its various iterations have certain limitations. In response, we carefully analyze the strengths and weaknesses of these natural vector methods and propose an improved version—an asymmetric covariance natural vector method (ACNV). This new method incorporates k-mer information alongside covariance computations with asymmetric properties between base positions. We tested ACNV on microbial genome sequence datasets, including bacterial, fungal, and viral sequences, evaluating its performance in terms of classification accuracy and convex hull separation. The results demonstrate that ACNV effectively captures sequence characteristics, showcasing its robust sequence representation capabilities and highlighting its elegant geometric properties.

1. Introduction

The size of genomic data continues to increase with advances in high-throughput sequencing technology and biology itself. As early as the last century, many computational biologists have done a lot of work on similarity comparisons between sequences, proposing a variety of methods including multiple sequence comparison (Vinga and Almeida, 2003). But the fact that sequence data themselves are of varying lengths and some sequences can be very long leads to an obvious efficiency bottleneck in analyzing and storing genomic data in the form of sequences. In addition, traditional sequence alignment algorithms such as MUSCLE (Edgar, 2004), ClustalW (Thompson et al., 1994), MAFFT (Katoh et al., 2002), have high time complexity and can be very time-consuming when faced with large-scale datasets. Therefore, the rapid construction of vectorized representations of biological sequence data of varying lengths that are easy to compute and store is of unique research value (Sarumi et al., 2024). For this reason, scholars have proposed many "alignment-free" methods to overcome the efficiency problems of sequence matching methods (Zielezinski et al., 2019).

A large number of alignment-free methods are aimed at constructing a vector representation of the DNA sequence for further analysis. This process can be described as an 'embedding' problem in the mathematical sense: consider the complete genome sequences of different organisms as a series of points in some space $\mathcal{G}$ called genome space, where a point represents a species. If there is a metric in this space, then this metric represents the evolutionary relationship between species, with the closer the distance the closer the evolutionary relationship between species. This is an idealized assumption, and in fact the space $\mathcal{G}$ does not exist for current researchers. We can only look for a 'measurable' space $\mathcal{D}$ and a mapping rule $f : \mathcal{G} \to \mathcal{D}$ such that: by means of the rule f, we can transform organisms in $\mathcal{G}$ into points in $\mathcal{D}$, and we can measure the closeness of evolutionary relationships between organisms by means of metrics defined on $\mathcal{D}$. Although evolutionary relationships between species can be analyzed more easily in space $\mathcal{D}$ using metric tools, the accuracy of the results of such analyses requires that the continuity of the mapping rule f is as good as possible, in other

* Corresponding authors.
E-mail addresses: drhu@bimsa.cn (G. Hu), yau@uic.edu (S.S.-T. Yau).
[1] These authors contribute equally to this work.

https://doi.org/10.1016/j.gene.2025.149532
Received 12 January 2025; Received in revised form 7 April 2025; Accepted 23 April 2025
Available online 12 May 2025

words that species that are close together in space $\mathcal{G}$ are also as close together in space $\mathcal{D}$ as possible.

In general, these sequence-based embedding methods can be categorized as follows based on how the embedding representation is computed (Zhou et al., 2024): **Sequence descriptors.** There are numerous descriptive-features-based methods for sequence analysis, leveraging various approaches such as word frequencies, the length of matching words, informational content between sequences, chaos game representation, and graphical representation of DNA sequences (Zielezinski et al., 2019). For instance, *Blaisdell* employed the k-mer model based on the classic string representation for genome sequence comparison (Blaisdell, 1986). *Qi* et al. introduced Composition Vector Tree (CVTree) using a composition vector approach (Qi et al., 2004). Kantorovitz utilized k-mer counts for comparing regulatory sequences. *Sims* et al. utilized feature (or k-mer) frequency profiles (FFP) of whole genomes for genome comparison (Sims et al., 2009). **Word2vec-like k-mer embeddings.** For example, *Patrick Ng* migrated the word2vec method from natural language processing to DNA and proposed the dna2vec method, which splits the sequence into k-mer fragments and then trains a neural network to construct the embedding representation (Ng, 2017), a similar approach is seen in Ren et al. (2022). *Han* et al. proposed a method of applying word2vec to k-mer information, and then selecting some dimensions to feed into a recurrent neural network via SVM (Han et al., 2022). However, word2vec-like approaches tend to suffer from longer model training times for k-mer embedding representations, which reduces their usefulness (Yu et al., 2023). **Foundation models.** With the great success of the large language model, this modeling paradigm began to spread to the front lines of research in a variety of fields and became known as the Foundational Model. A growing number of base models have also emerged in the field of sequence comparison. For example, *Ji* et al. in analogy to bert, proposed a bi-directional encoder-based pre-training model for DNA sequence encoding (Ji et al., 2021), and *Zhou* et al. strengthened the inter-species discrimination ability of the resulting embedding vectors by employing a comparative learning paradigm based on DNABert-2 (Zhou et al., 2023) to obtain DNABert-S (Zhou et al., 2024).

Each of these methods has its own problems, for example, methods based on sequence descriptive quantities are relatively simple and straightforward, but it is often difficult to extract information in a more comprehensive way, thus leading to a limitation of their descriptive power. Methods similar to word2vec act as a prequel to the base model class of methods, and tend to be less effective and generalizable than the base model while paying a price in training time. As for base models, some studies have found that they do not perform so well in some cases either, and they tend to be relatively devoid of biological interpretability (Zhou et al., 2024). Although there have been some modeling studies that emphasize interpretability (Elmarakeby et al., 2021; van Hilten et al., 2021, 2024), work in this area has been primarily directed at somewhat more specific genomics tasks, and the interpretability is more in the model (e.g., model weights) than in the embedding vectors. To summarize, despite the existence of a wide variety of approaches, it is still problematic, from both a mathematical and a biological point of view, how to implement vector representations of genome sequences in a way that combines both elegant mathematical properties and good biological interpretability to effectively distinguish sequences of different organisms.

Descriptor-based methods tend to have the best interpretability, but their weakness usually lies in the fact that the limited manual setting leads to not enough information being accommodated (Zhou et al., 2024). Among the descriptor-based approaches, there is a class of methods named natural vector methods that establish a relatively well-developed framework. *Deng* et al. proposed the natural vector method for biological sequences in 2011 (Deng et al., 2011), providing a theoretical framework for a unified vectorized representation of genome sequences. The method is computationally simple and scalable, and it has clear biological significance and sound mathematical properties. Subsequent further research has developed several variants of this method, making the natural vector method more and more sophisticated. For example, the k-mer natural vector method proposed by *Wen* et al. in 2014 introduces k-mer information into the computation of natural vectors (Wen et al., 2014). In addition, it has been shown that a proper choice of the method of calculating the natural vectors can lead to the so-called convex hull principle: the convex hulls formed by the natural vectors of the genome sequences of differently categorized organisms are non-intersecting with each other (Tian et al., 2018). This suggests that the natural vector method has good properties from the geometric point of view and the potential to be further investigated. Against this background, we are more interested in the apparent simplicity (computationally efficient) and interpretability (clear biological significance) of this kind of descriptive characterization based methods.

Another class of sequence comparison methods does not construct a vectorized representation of the sequences, but rather computes the similarity or distance matrix between the sequences directly. For example, Co-phylog (Yi and Jin, 2013) generates 'microcomparisons' by using 'context' at each 'object' of a sequence to estimate evolutionary distances, while andi (Haubold et al., 2015) uses an augmented suffix array to detect pairs of maximal unique matches, and then counts the number of substitutions at each position to estimate the evolutionary distance between genomes. Some of the new comparison-free methods proposed recently also fall into this category, such as the PEAFOWL (Zahin et al., 2025) method, which encodes the presence or absence of k-mers in a genome sequence into a binary matrix and uses maximum likelihood to estimate a phylogenetic tree, and the TF–IDF method (Delibaş, 2025), which represents DNA sequences as n-grams and then applies the Term Frequency–Inverted Document Frequency (TF–IDF) of the Natural Language Processing algorithm to construct the sequence similarity matrix. While such methods can be designed from different perspectives of sequence similarity comparison (e.g., homology, context matching), they lack the possibility of exploring further mathematical properties and their utility as sequence encoding plugins, as they do not directly construct a mathematical representation of the sequence.

1.1. Statistical-descriptors-based representation

The use of statistical descriptors to compress sequence information is a very common tool to build representation of DNA/RNA sequences in bioinformatics (Fan et al., 2015; Lu et al., 2017; Chan et al., 2014; Murray et al., 2017). These descriptors can both form a vector representation of DNA/RNA sequences or serve as features for machine learning algorithms. We mainly consider the former scenario, when the main challenge is to compress as much sequence information as possible in an interpretable form in a rather limited number of vector dimensions.

Let us start with a few basic examples. Taking DNA as an example, and considering it as a language, then a genome sequence can be considered as a long sentence, and after determining the word list, we can construct statistical descriptors based on certain probability distributions of the word list. Let $S = s_1 s_2 \cdots s_n$ be a DNA sequence of length n, $s_i \in L$, where L is the word list (e.g., $L = \{A, C, G, T\}$). Let $e_l(\cdot) : L \to \{0, 1\}$ be an indicator function such that $e_l(s_i) = 1$ if $s_i = l \in L$ and 0 if not. Then we can construct some statistical descriptors around the distribution of elements in the word list:

- Frequency of occurrence of words. Define

$$n_l = \sum_{i=1}^{n} e_l(s_i), \quad \forall\, l \in L, \tag{1}$$

  then n_l represents the occurrence time of word $l \in L$.
- Positional distribution of words.

1. Mean. Define

$$\mu_l = \sum_{i=1}^{n} i \cdot e_l(s_i), \quad \forall\, l \in L, \tag{2}$$

then μ_l represents the average occurrence position of word $l \in L$.

2. Variance. Define

$$D_l = \frac{1}{n_l} \sum_{i=1}^{n} (i - \mu_l)^2 \cdot e_l(s_i), \quad \forall\, l \in L, \tag{3}$$

then D_l represents the variance of the occurrence position of word $l \in L$.

The above definition does not restrict word lists, so the above statistical descriptors apply to any meaningful word list, allowing the derivation of additional descriptors. In particular, the word list commonly used in genome sequence analysis is k-mer, where k takes a positive integer. We denote the k-mer word list as L_k.

Natural vectors

Natural vector (basic form). Natural vector method was firstly developed by *Deng* et al. in 2011 (Deng et al., 2011). Taking the word list as L_1 and combining the statistical descriptors above gives the most basic version of the natural vector which is a 12-dimensional vector:

$$\boldsymbol{v} = (n_A, n_C, n_G, n_T, \mu_A, \mu_C, \mu_G, \mu_T, D_2^A, D_2^C, D_2^G, D_2^T)', \tag{4}$$

where D_2^l is the modified variance:

$$D_2^l = \frac{1}{n} D_l = \frac{1}{n n_l} \sum_{i=1}^{n} (i - \mu_l)^2 \cdot e_l(s_i). \tag{5}$$

The introduction of sequence length as an additional scaling factor is intended to reduce the size of this item.

Natural vector with higher-order moments. If we do not limit ourselves to second-order moment statistics, then we can expand the dimension of the basic natural vectors by the size of the word list and the order of the highest-order moment. Define the high-order components of natural vector as

$$D_p^l = \frac{1}{n^{p-1} n_l^{p-1}} \sum_{i=1}^{n} (i - \mu_l)^p \cdot e_l(s_i). \tag{6}$$

Then the natural vector with high-order moments is given by

$$\begin{aligned} \boldsymbol{v}_h = (&n_A, n_C, n_G, n_T, \mu_A, \mu_C, \mu_G, \mu_T, D_2^A, D_2^C, D_2^G, D_2^T, \\ &\cdots, D_m^A, D_m^C, D_m^G, D_m^T)'. \end{aligned} \tag{7}$$

K-mer natural vector. If we do not restrict ourselves to the most basic word list $L = \{A, C, G, T\}$, but take as words strings of nucleotide bases of length k (k-mer), then k-mer natural vectors can be defined in a similar way to basic natural vectors. Specifically, denote the word list as $L_k = \{l_1, l_2, \ldots, l_{4^k}\}$ and define indicating function

$$e_{l_i}(j) = \begin{cases} 1 & \text{if } s_j s_{j+1} \cdots s_{j+k-1} =: s_{j:j+k-1} = l_i \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

Then we can define the components of the k-mer natural vector after the basic natural vector as follows:

- K-mer occurrence:

$$n_{l_i} = \sum_{j=1}^{n-k+1} e_{l_i}(j). \tag{9}$$

- K-mer positional mean:

$$\mu_{l_i} = \frac{1}{n_{l_i}} \sum_{j=1}^{n-k+1} i \cdot e_{l_i}(j). \tag{10}$$

- K-mer positional variance (or high-order moments):

$$D_p^{l_i} = \frac{1}{(n-k+1)^{p-1} n_{l_i}^{p-1}} \sum_{j=1}^{n-k+1} (j - \mu_{l_i})^p \cdot e_{l_i}(j). \tag{11}$$

Notice that a sequence of length n has only $n-k+1$ consecutive k-mer fragments, so the upper bound of the summation in the above equations is $n-k+1$. Then one can define the k-mer natural vector as

$$\boldsymbol{v}_k = (n_{l_1}, n_{l_2}, \ldots, n_{l_{4^k}}, \mu_{l_1}, \mu_{l_2}, \ldots, \mu_{l_{4^k}}, D_2^{l_1}, D_2^{l_2}, \ldots, D_2^{l_{4^k}})'. \tag{12}$$

Natural vector with covariance. There is a limitation of only raising the order of moments. In fact, the moments of each order, including mean and variance, mainly reflect the characteristics of the positional distributions of individual words, but do not reflect the correlation of the positional distributions among different words. The covariance, as a commonly used statistic, can reflect the similarity of the trend of change between two variables, and is very suitable as a supplement to the elements in the natural vector. In the field of bioinformatics, covariance has been more widely used (Price, 1970; Shen and Li, 2016), but it is not common to use it directly for the inscription of word distribution information in sequences. To compute the covariance for the distributional positions of words in the sequence, there are several problems to overcome. The first is that the classical definition of a discrete covariance requires that the two discrete variables have the same number of samples, but this may not be satisfied in the case of DNA sequences. Secondly we also need to make the covariance compatible with the previously defined variance terms.

Zhao et al. first introduced covariance into the natural vector method in 2018 (Tian et al., 2018). In order to overcome the constraint that classical discrete covariance imposes on the set of values taken by two variables to be of the same size, they propose an averaging approach to compute the covariance for the positional distributions of two different words: suppose the position sets of word l_1 is $A = \{a_1, a_2, \ldots, a_m\}$ and that of l_2 is $B = \{b_1, b_2, \ldots, b_n\}$ where $m > n$, and the sequence length is N, then

$$\begin{aligned} Cov(l_1, l_2) &= \frac{1}{N \cdot C_m^n} \sum_{A_* \subset A, |A_*| = n} Cov(A_*, B) \\ &= \frac{1}{N \cdot C_m^n} \sum_{A_* \subset A, |A_*| = n} \frac{1}{n} \sum_{j=1}^{n} (a_j^* - \mu_{l_1}^*)(b_j - \mu_{l_2}), \end{aligned} \tag{13}$$

where $a_j^* \in A_*$ indicates the sampled n positions from original position set A of the word l_1, and $\mu_{l_1}^*$ is the corresponding mean position. Based on this way of definition, it both generalizes the classical discrete covariance so that it can be applied to the positional distributions of two different words in a sequence, and is compatible with the variance, i.e., the above definition is equivalent to the original definition of the variance when $l_1 = l_2$. However, the calculations in this way of definition are relatively cumbersome, and this calculation does not facilitate a biological interpretation of its meaning.

Sun et al. in 2022 proposed a new way of defining the covariance between word distributions in a sequence (Sun et al., 2022), which is significantly simplified in form compared to the previous definition:

$$Cov(l_1, l_2) = \frac{1}{n \sqrt{n_{l_1} n_{l_2}}} \sum_{i=1}^{n} (i - \mu_{l_1})(i - \mu_{l_2}) \cdot e_{l_1 l_2}(s_i), \tag{14}$$

where $e_{l_1 l_2}(s_i) = 1$ if $s_i = l_1$ or $s_i = l_2$, otherwise $e_{l_1 l_2}(s_i) = 0$. The covariance term is added to the basic natural vector so that an 18-dimensional vector is obtained:

$$\begin{aligned} \boldsymbol{v}_c = (&n_A, \ldots, n_T, \mu_A, \ldots, \mu_T, D_2^A, \ldots, D_2^T, Cov(A, C), Cov(A, G), \\ &Cov(A, T), Cov(C, G), Cov(C, T), Cov(G, T))'. \end{aligned} \tag{15}$$

This definition has the same symmetry as the classical covariance and shows a clean and beautiful form. *Sun* et al. applied this method to the classification study of some microbial datasets and found that the introduction of the covariance led to a significant improvement in

the classification of the traditional natural vector method on these datasets (Sun et al., 2022). In addition, the authors also point out in the paper that the method can be further defined for k-mer, e.g., for 2-mer, $Cov(AA, AC)$ can be computed in one unit of two consecutive positions using almost exactly the same formula. However, when this approach is really expanded in combination with k-mer, the dimensionality increase is on the level of the square of the number of k-mer fragments, and it is easy to be confronted with extremely high variable dimensions (e.g., defining covariance for 2-mer introduces an additional $16 * 16 = 256$ dimensions).

To summarize, the introduction of the covariance term can bring new information to the natural vectors besides the descriptive statistics of "individual word position distribution" such as the moments of each order, which is beneficial to the application of the natural vector method, but there are some problems with the existing methods.

1.2. Convex Hull principle

Convex hull. The continuity of the mapping from genome space $\mathcal{G}$ to vector space $\mathcal{D}$ cannot be described directly and can often only be judged by flanking. For example, evaluation metrics based on genome classification tasks (e.g., accuracy, F1 score, etc.), and evaluation metrics based on genome clustering tasks (e.g., purity, ASI index, etc.). But let us focus on the embedding vectors themselves, and if there exists a natural mathematical structure such that the differences between the embedding vectors of different kinds of organisms are naturally described by that structure, then the effectiveness of this embedding vector approach can be demonstrated to some extent. The smallest convex set containing a set of points is called the convex hull formed by this set of points. More precisely, let $P = \{\boldsymbol{p}_1, \boldsymbol{p}_2, \dots, \boldsymbol{p}_m\}$ with $\boldsymbol{p}_i \in \mathbb{R}^k$ be a point set in Euclidean space, then the convex hull spanned by this point set is defined by

$$\begin{aligned}\mathrm{Conv}(P) = \{&\lambda_1\boldsymbol{p}_1 + \lambda_2\boldsymbol{p}_2 + \cdots + \lambda_m\boldsymbol{p}_m \mid \boldsymbol{p}_i \in P,\\ &0 \le \lambda_i \le 1, \forall\, 1 \le i \le m, \text{ and } \lambda_1 + \lambda_2 + \cdots + \lambda_m = 1\}.\end{aligned} \tag{16}$$

In the usual Euclidean space, a convex hull can be visualized as a convex polyhedron (convex polygon). In two-dimensional Euclidean space, a convex hull can be visualized as a ring of rubber bands enclosing a number of nails nailed to a flat wooden board.

Convex hull principle. As the saying goes, "Birds of a feather flock together", if there is some commonality between a set of genomic sequences, then the corresponding "nails" should be nailed to similar areas of the plank, while the "rubber band surroundings" formed by different types of genomic sequences should not overlap each other. This intuition corresponds to the so-called convex hull principle. *Tian* et al. proposed the convex hull principle of the natural vector method in 2018 (Tian et al., 2018), which states that the convex hulls formed by the natural vectors of the genome sequences of organisms belonging to different classes (e.g., families or genus) taxonomically do not intersect with each other, and that the specific choices of the natural vectors here may vary somewhat with the type of organisms (e.g., the natural vectors with higher-order moments of different orders) (Wang et al., 2019; Zhao et al., 2019; Sun et al., 2021). In non-mathematical language, the convex hull principle based on the natural vector method is like placing the genome sequence data of organisms in a small universe, where each sequence of organisms is analogous to a star, and the convex hull is a nebula formed by the stars of a class of organisms, and the convex hull principle suggests that these nebulae are separated from each other. If this principle always holds true, then we can even look for unknown stars from known nebulae, in other words, for as-yet-undiscovered genome sequences of known types of organisms. This theoretical paradigm clearly has applications (e.g., in the case of high-variability epidemic outbreaks), and there is already a body of work devoted to theorizing about it (Zhao et al., 2020; Jiao et al., 2021).

2. Results

2.1. Natural vector with asymmetric covariance

Limitations of existing natural vector forms. Despite the fact that there are already many members of the natural vector family, there are still some limitations of such methods. For example, while natural vectors with covariance complement the statistical correlation between the positional distributions of different words, and k-mer natural vectors complement the distributional information of k-mer fragments, the two approaches are not yet well compatible. This is because if the covariance is defined directly for k-mer, on the one hand the interpretability is rather limited (in other words, not sufficiently biologically motivated), and on the other hand this would cause the vector dimension to grow rapidly, which would be inconvenient for practical use. In view of the above, we would like to propose a scheme to combine the properties of several different forms of natural vectors above.

Moreover, in most of the previous work on the convex hull principle, natural vectors with higher-order moments are used to construct convex hulls, but the higher-order moments portion of the vectors exhibits significant numerical degeneracy with increasing order (i.e., it is very close to zero compared to the first number of dimensional components of the vectors), which may introduce additional computational errors to the test of the convex hull principle. Therefore, we hope to overcome the above problems encountered in the traditional natural vector dimensioning process in a new way.

For natural vector with covariance, the covariance term is added mainly to reflect the statistical correlation of the positional distributions of different words, there is no qualification imposed on the positional distributions here, and the two words used to calculate the covariance are computationally equivalent, so we can call it a 'natural vector with symmetric covariance'. In k-mer natural vectors, k-mer fragments, which are widely used in bioinformatics, are chosen as word lists to construct natural vectors, which leads to a natural expansion of vector dimensions and the amount of embedded information. Then, we can consider combining the k-mer and covariance elements, e.g., by defining the '2-mer covariance', which means the computation of covariance is conditioned on the 2-mer composed by two specific letters. Based on these considerations, we have drawn on the properties of existing natural vector forms to devise a new way of calculating the covariance.

Introduction of asymmetric covariance. In equation [wq], the covariance term is defined as

$$Cov(A, C) = \frac{1}{n * \sqrt{n_A} * \sqrt{n_C}} \sum_{s_i \in \{A,C\}} (i - \mu_A)(i - \mu_C). \tag{17}$$

It is easy to see that the positions of A,C can be exchanged in this equation without changing the value of the equation, i.e. it is symmetric. If we want to incorporate the k-mer information, for example, we can then add a '2-mer condition' to the result of this unconditional calculation: the letters involved in the covariance calculation must appear in the 2-mer $AC(\in L_2)$. Then we get a 'complete 2-mer condition' version of the asymmetric covariance:

$$\begin{aligned}Cond - ACov(A, C) = &\frac{1}{n * \sqrt{n_{A|AC}} * \sqrt{n_{C|AC}}}\\ &\sum_{s_i s_{i+1} = AC \text{ or } s_{i-1} s_i = AC} (i - \mu_{A|AC})(i - \mu_{C|AC}),\end{aligned} \tag{18}$$

where the conditional operator "$|AC$" means to restrict the calculation post-fixed by this operator to all 2-mer AC's in the sequence. However, in this way of definition, we completely discard the positions of letters of the same type appearing outside the 2-mer in the sequence, which may bring about incomplete information. In fact, we have the following observations:

$$n_{A|AC} = n_{C|AC} = n_{AC}, \quad \mu_{C|AC} = \mu_{A|AC} + 1 = \mu_{AC} + 1, \tag{19}$$

where n_{AC} and μ_{AC} are both terms defined according to the description of k-mer natural vectors in the previous subsection. However, by a simple mathematical derivation we find that this fully conditional form does not give us new information compared to the traditional k-mer natural vector components, which suggests that it is not desirable to rigidly add the k-mer condition to the covariance formula:

$$\begin{aligned} &Cond-ACov(A,C)\\ &=\frac{1}{n*n_{AC}}\Big[\sum_{s_is_{i+1}=AC}(i-\mu_{AC})(i-\mu_{AC}-1)+\\ &\qquad\sum_{s_is_{i+1}=AC}(i+1-\mu_{AC})(i+1-\mu_{AC}-1)\Big]\\ &=\frac{1}{n*n_{AC}}\sum_{s_is_{i+1}=AC}(i-\mu_{AC})[(i-\mu_{AC}-1)+(i+1-\mu_{AC})]\\ &=\frac{2}{n*n_{AC}}\sum_{s_is_{i+1}=AC}(i-\mu_{AC})^2\\ &=2*D_2^{AC}. \end{aligned} \tag{20}$$

Similar to [eq], it can be shown to hold for the general k-mer case, except that the final product coefficient changes from 2 to k. Thus, if the full k-mer condition is added directly to the covariance, the covariance degenerates into the variance of the k-mer position, which defeats the idea of combining information from both the k-mer and the covariance.

Inspired by this, the asymmetric covariance that we are trying to propose takes a "non-complete k-mer condition" approach. To be more precise, we adopt a combination of local information (the distribution of the positions of the letters in the constraints of the 2-mer condition) and global information (the distribution of the positions of all the individual letters in the sequence):

$$ACov(A,C)=\frac{1}{n*\sqrt{n_A}*\sqrt{n_C}}\sum_{s_is_{i+1}=AC\ \text{or}\ s_{i-1}s_i=AC}(i-\mu_A)(i-\mu_C). \tag{21}$$

In short, we let the local conditional operator act only on the individual letter positions involved in the summation, while retaining the global computation of the two statistics, the number of letters appearing in the equation and the average letter position. This combination of local k-mer word position conditioning and global mean word position avoids degradation of the covariance itself (unless the words in the sequence that participate in the covariance computation appear only in the k-mer used as conditioning).

Numerical example. To make it clearer, let us consider a simple example. Let $S=ACCTGAC$ be a DNA sequence, we focus on the covariance of letter A and letter C. Clearly we have $n=7$ and

- $n_A=2$ and $n_{A|AC}=2$;
- $n_C=3$ but $n_{C|AC}=2$;
- $\mu_A=(1+6)/2=3.5$ and $\mu_{A|AC}=(1+6)/2=3.5$;
- $\mu_C=(2+3+7)/3=4$ but $\mu_{C|AC}=(2+7)/2=4.5$.

so one can calculate that

$$Cov(A,C)=\frac{1}{7\sqrt{6}}\sum_{i\in\{1,2,3,6,7\}}(i-3.5)(i-4), \tag{22}$$

$$Cond-ACov(A,C)=\frac{1}{14}\sum_{i\in\{1,2,6,7\}}(i-3.5)(i-4.5), \tag{23}$$

$$ACov(A,C)=\frac{1}{7\sqrt{6}}\sum_{i\in\{1,2,6,7\}}(i-3.5)(i-4). \tag{24}$$

The third equation above is our proposed asymmetric covariance term. We can figuratively say that 'the local information brought about by the 2-mer condition is all under the summation symbol'. The asymmetry here stems directly from the ordering of the 2-mer condition itself, i.e., AC and CA as 2-mer are distinct fragments. Clearly one can see that

$$\begin{aligned} ACov(A,C)&=\frac{1}{7\sqrt{6}}\sum_{i\in\{1,2,6,7\}}(i-3.5)(i-4)\approx 68.89,\\ ACov(C,A)&=\frac{1}{7\sqrt{6}}\sum_{i\in\emptyset}(i-3.5)(i-4)=0. \end{aligned} \tag{25}$$

Higher-order forms. Moreover, since k-mer is not restricted to be 2-mer, we can further define the asymmetric (generalized) covariance between k letters. Let $l_j=(t_1t_2\cdots t_k)\in L_k$ be an arbitrary k-mer, then

$$\begin{aligned} &ACov(t_1,t_2,\ldots,t_k)\\ &=\frac{1}{n\prod_{s=1}^k\sqrt{n_{t_s}}}\sum_{i\in\boldsymbol{Pos}_{k,l_j}}(i-\mu_{t_1})(i-\mu_{t_2})\cdots(i-\mu_{t_k})\\ &=\frac{1}{n\prod_{s=1}^k\sqrt{n_{t_s}}}\sum_{p=1}^{k}\sum_{i=1}^{n_{t_p|l_j}}\prod_{p=1}^{k}(i-\mu_{t_p}), \end{aligned} \tag{26}$$

where $\boldsymbol{Pos}_{k,l_j}$ consists of each position of all instances of all occurrences of $l_j\in L_k$ in the sequence S.

After adding the components of the asymmetric covariance to the first eight base dimensions of the natural vector, we get a 24-dimensional vector. In general, adding higher order asymmetric covariance components gives us a vector of dimension

$$8+4^2+4^3+\cdots+4^k=8+\frac{1}{3}(4^{k+1}-16)=\frac{1}{3}(4^{k+1}+8), \tag{27}$$

where k corresponds to the local k-mer condition chosen for the computation of the highest order asymmetric covariance component in the vector. We point out again that the additional dimensionality associated with the extension of their proposed covariance over k-mer in Sun et al. (2022) would be of the order of the square of the above equation.

2.2. Convex Hull principle test

Convex hull principle. For each dataset, we first compute the vector representations of all the sequences, and then determine whether the two convex hulls in any convex hull pair intersect or not, here a convex hull corresponds to a family in taxonomy. According to the article that proposes the convex hull principle, the number of pairs of intersecting convex hulls should get smaller and smaller as the dimensionality of the natural vector rises until it goes to 0, and thus the convex-hull principle is fully established.

It should be noted that, as an empirical rule, the convex hull principle does not guarantee that any vector representation will satisfy the convex hull principle for any dataset, but if a vector representation can bring the number of intersecting convex hull pairs to a known minimum, or even to 0 in order to truly establish the convex hull principle, then we can assume that such a vector representation can be very effective for the refinement of sequence information, and has a good ability to classify sequences. We can understand this from a different perspective. In the transformer structure widely used in modern deep learning, what the attention module is doing is also "constructing" a convex hull of intermediate-level vector representations (values) based on the input content, and then convexly combining (attention scores) the elements in it to generate vector representations of more elements, and the training process of this model can be regarded as a loss-driven optimization of the "semantic convex hulls". Thus, there is good reason to use the convex hull principle to test the validity of the vector representation. Starting from the convex hull principle, as a measure of the validity of the vector representation, we can define the ratio of the number of disjoint convex hull pairs to the total number of convex hull pairs under a vector representation as the "convex hull principle validity ratio" for each dataset. The results of convex hull principle test is displayed in Table 1.

Table 1
Convex hull principle validation ratio on 3 datasets. DCH: number of disjoint convex hull pairs; CHPVR: convex hull principle validation ratio.

Dataset	Indicators	Dimension of ACNV (with k-mer condition)			
		24 (2-mer)	88 (3-mer)	344 (4-mer)	1368 (5-mer)
Fungi	DCH	99,986	108,400	108,810	108,811
	CHPVR	91.89%	99.62%	99.99%	100.00%
Virus	DCT	3384	3400	3403	
	CHPVR	99.44%	99.91%	100.00%	
Bacteria	DCH	15,746	15,753		
	CHPVR	99.96%	100.00%		

Fungi DNA barcodes. The Fungi DNA barcodes dataset has a total of 467 distinct families, corresponding to $C_{467}^2 = 108811$ convex hull pairs. As a comparison, in Tian et al. (2018), their dataset contains 448 families, corresponding to $C_{448}^2 = 100128$, while in Sun et al. (2022), their dataset contains 467 families. Our dataset is much closer to Sun et al. (2022) (this is mainly due to the fact that Sun et al. (2022) was more updated and the version of the database is much closer to the current version), so we compare our results with (Sun et al., 2022). In their work, they tried both classical 12-dimensional natural vectors and 18-dimensional natural vectors with covariance, and for all 108,811 convex hull pairs, they obtained 75,237 and 88,719 disjoint convex hull pairs in both cases, which correspond to 69.14% and 81.53% of the convex hull principle validity ratio, respectively. We performed tests using the proposed asymmetric covariance natural vectors (ACNV) and finally achieved a 100% verification ratio of the convex hull principle.

Viral genome sequences. The viral genome sequences dataset has a total of 83 distinct families, corresponding to $C_{83}^2 = 3403$ convex hull pairs, which is the same as (Sun et al., 2022). In their work, the classical natural vectors and natural vectors with covariance give 3321 and 3322 disjoint convex hull pairs, corresponding to 97.59% and 97.62% validation ratios of the convex hull principle, respectively. With our proposed ACNV, it is still possible to achieve a 100% validation ratio of convex hull principle can also be achieved.

Bacterial genome sequences. The bacterial genome sequences dataset has a total of 178 distinct families, corresponding to $C_{178}^2 = 15,753$ convex hull pairs, which is the same as (Sun et al., 2022). In Sun et al. (2022), the classical natural vectors and natural vectors with covariance give 14,565 and 15,160 disjoint convex hull pairs, corresponding to 92.46% and 96.24% validation ratios of the convex hull principle, respectively. And with the use of our proposed ACNV, a 100% validation ratio of convex hull principle can also be achieved (see Table 1). In summary, it can be seen that based on ACNV, we can realize the complete convex hull principle for three different microbial genome sequence datasets, which surfaces that ACNV has obvious effectiveness in extracting sequence information, especially with the beautiful convex separation geometric property in the vector representation space.

2.3. Bacterial genome sequence classification

Of the three datasets used for the convex hull principle experiments, the bacteria dataset has the widest range of sequence lengths and is relatively rich in the number of sequence families, making it one of the more challenging datasets from a sequence categorization perspective. In order to more directly reflect the potential of ACNV for sequence classification, we apply them to the sequence classification task on the bacterial genome sequence dataset by combining them with MLP and XGBoost, two common machine learning classification models. In both experiments, we used only 24-dimensional asymmetric covariance natural vectors (i.e., under the 2-mer condition).

Table 2
Classification model training settings. BS: batch size; LR: learning rate; Opt: optimizer; MD: max depth.

MLP	Epochs	BS	LR	Opt
	1000	16	0.0005	Adam
XGBoost	# of Estimators	MD	LR	
	250, 350, 500, 550	5, 20, 40, 60	0.05, 0.10, 0.15	

MLP model settings. We use a simple MLP model to perform sequence classification task. The model consists of four hidden layers and a softmax classification layer. The dimensions of the four hidden layers are 1024, 512, 256, and 128, and all of them use the ReLu activation function, layernorm, and dropout of 0.5. The 16373 sequences in the dataset are divided into training, evaluation, and test sets in the ratio of 75:20:5. The other settings of the training process are listed in Table 2

XGBoost model settings. We used the XGBoost library to create the classifiers and GridSearchCV to perform a grid search on the hyperparameters of the model to find the best combination of hyperparameters. The dataset is divided into training, validation and test sets in the ratio of 90:9:1 after removing labels with less than 100 sequence entries, leaving 13135 sequences. Further information on model hyperparameters is given in Table 2.

Classification results. The trained MLP model and the best XGBoost model obtained through parameter search were used to test the classification accuracy on the test set. The predicted labels of the prediction set predicted after model training are compared with the correct labels and the proportion of labels predicted accurately is calculated. The MLP model gave 96% classification accuracy on the test dataset consisting of 819 sequences, while the XGBoost model gave 98% classification accuracy on the test dataset consisting of 132 sequences. In contrast, classical 12-dimensional natural vector, 18-dimensional natural vector with covariance, and 48-dimensional 2-mer natural vector achieved classification accuracies of 72%,90%, 81%,90% and 89%,93%, respectively. This shows that ACNV has better representation ability than the other three natural vector methods. In order to see the potential of ACNV in sequence classification more graphically, we used UMAP and tSNE to visualize the dimensionality reduction of the ACNV corresponding to the labels of the top five sequence counts, and it can be seen from the figure that the ACNV distinguishes different types of sequences very well: (see Fig. 1).

In these two experiments, we have still only used a 24-dimensional ACNV, and the results may be further improved if the model is trained with an ACNV with higher-order k-mer conditions. We believe this demonstrates the potential of ACNV as a powerful tool for sequence classification/clustering tasks.

3. Discussion

Non-degeneracy of ACNV components. One of the biggest improvements of ACNV over other versions of the natural vector method is the ability to combine covariance with k-mer information without losing information and keeping the values of the individual components from varying too much. This is particularly useful when verifying the convex hull principle, since numerical instability of the vector components can seriously affect the correctness of the results of the convex hull intersection determination algorithm. Most of the validation of the convex hull principle in previous work has used natural vector representations with higher-order moments (Tian et al., 2018; Zhao et al., 2019; Sun et al., 2021), which carry indices of sequence length and word frequency as denominators in their definitions that can lead to the phenomenon of degeneracy of a large number of vector components after a certain order - i.e., the appearance of values that are almost zero. If the k-mer natural vectors or natural vectors with covariance are used to

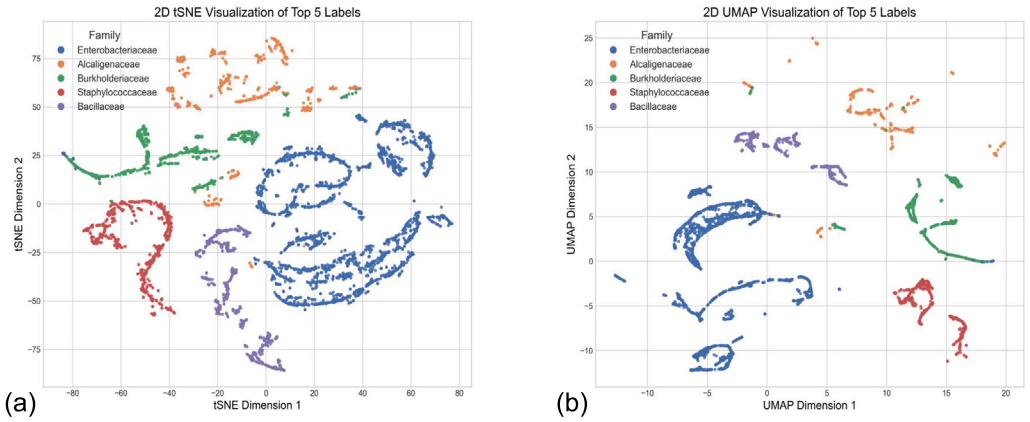

Fig. 1. T-SNE and UMAP visualization of the ACNV corresponding to the labels of the top five sequence counts.

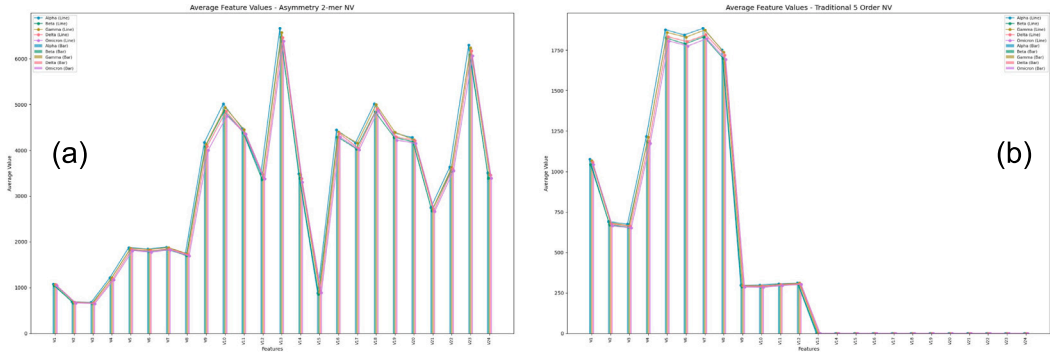

Fig. 2. Line and bar plots of the average values of the vector components, each dimension is displayed by averaging the computed results over all 3411735 sequences. (a) ACNV under 2-mer condition (dim=24). (b) Natural vector with 5-order moments (dim=24). It is clear that the higher-order moment part of the natural vector has significant numerical degeneracy, whereas ACNV has no such problem.

construct the convex hull principle, it is often limited by the fact that the vectors contain incomplete information leading to problems such as incomplete separation of the convex hull pairs or high dimensionality (e.g., when trying to combine the k-mer with the covariance). ACNV, on the other hand, can introduce the covariance information between word distributions while keeping the same dimension as the k-mer natural vectors, which overcomes the lack of complete information and also avoids the degradation of the vector components.

We tested the SARS-CoV-2 spike gene nucleotide sequence dataset consists of 3411735 sequences. It can be seen in Fig. 2(b) that if a 24-dimensional natural vector with 5-order moments used, the components from the 13th dimension onwards begin to differ significantly in numerical magnitude from the first 12 dimensions, resulting in a degenerate effect. However, in Fig. 2(a), the differentiation effect of each dimension is maintained by ACNV, so we say that ACNV has non-degeneracy in vector components. And the numerical non-degeneracy of the vector components can still be guaranteed even in the 1368-dimensional ACNV, as is shown in Fig. 3. We speculate that this is one of the reasons why ACNV is able to realize the fully convex hull principle in our experiments. In addition, we note that the value of the 15th dimension covariance in Fig. 2(a) is significantly smaller compared to the other covariance dimensions. The 2-mer sequence corresponding to this dimension is "CG", and this small value may indicate a relatively weak preference for codons containing "CG" in the spike gene sequence of the SARS-CoV-2, which is supported by other studies (Fumagalli et al., 2023).

Further research directions. Based on the good representation ability and beautiful geometric properties of ACNV, we believe that some of the following further studies (applied or exploratory) can be carried out based on ACNV:

- As a plug-and-play tool for any sequence-comparison type task: based on ACNV's fast algorithms and sequence representation capabilities, we can use ACNV to quickly construct a uniform dimensional vector representation of biological sequence data of any length, which is plug-and-play for all kinds of sequence comparison tasks. From the perspective of convenient application, for human/mouse latest version gene database, we have calculated the ACNVs for gene sequences, and can provide the service or online dynamic calculating of ACNV upon input an fasta gene sequence in our upcoming releases. For example, natural vectors can be used as data encoders in deep learning models based on biological sequences, which may be able to accelerate model convergence.
- Construction of gene correlation network diagram based on ACNV: by calculating the ACNVs of the gene sequences, and then further calculating the correlation between the ACNVs (e.g., by Pearson correlation coefficient, etc.) to form a correlation matrix, we can use the ACNVs as the points and the correlation matrix as the edges to construct a gene association network, which can provide a basic tool for the study of gene interactions and other aspects.
- Solving the vector-to-sequence problem via the convex hull principle under ACNV: since ACNV has the potential representation capability to guarantee that the convex hull principle holds, we can first construct a convex hull using ACNV for the sequence dataset of interest, and then search for new ACNV points in the convex hull, solving the inverse problem of ACNV-to-seq for the purpose of searching for new sequences that have not yet been discovered. There is some theoretical difficulty in this application because the scattering points within a convex packet in a high-dimensional space tend to be distributed at the edges

Fig. 3. Line and bar plots of the values of the components of ACNV with 5-mer conditioned asymmetric covariance (dim=1368), each dimension is displayed by averaging the computed results over all 3411735 sequences. (a) First 344 dimensional components; (b) 345th to 1368th dimensional components. It can be seen that the introduction of higher dimensions does not lead to a significant numerical degradation effect either.

of the packet, and thus it is not easy to find new points in it. Deep learning based methods usually produce high dimensional embedding representations and have not been verified for the convex envelope principle. However, ACNV combines the geometric property of the convex packet principle with the ability to represent it efficiently in low dimensions, and thus has the potential to be applied to this problem.

4. Conclusions

In this paper, we introduce asymmetric covariance natural vector (ACNV), an improved version of natural vectors. This method firstly overcomes the problem that natural vectors with higher-order moment statistics alone cannot imply the correlation of the positional distributions of different words. Secondly, it combines the natural vectors with covariance and the k-mer natural vectors in a form of "k-mer conditioned", so that ACNV can combine the information of covariance between the positional distributions of words and the k-mer fragments of the words while maintaining the same dimensionality as the k-mer, which further overcomes the problem that the traditional covariance natural vector method is not well compatible with k-mer. Through experiments on microbial (bacterial, fungal, and viral) genome sequence datasets, ACNV has achieved 100% validation rate of the convex hull principle, which reflects the beautiful geometric properties of the ACNV method in constructing vector representations of sequences and its excellent ability to characterize the differences of biological taxa, and also provides a new tool for subsequent research on biological sequence data.

5. Materials and methods

Methods. See the first subsection of Results, 'Natural Vector with Asymmetric Covariance', for an analysis of the definition of ACNV; see the third subsection of Results for a description of the training and evaluation of the sequence classification models covered in the paper. For a description of the training and evaluation of the sequence classification model described in the paper, see Section 3 of the Results, 'Bacterial Genome Sequence Classification'.

Materials. Our method is based on the development of the natural vector method, and the most direct way to test the capability of the method itself is to make a direct comparison with other natural vector methods that have been previously proposed on the same dataset. To this end, we fused the datasets used in the two previously mentioned prior works that introduced covariance into the natural vector method, and tested our proposed new method on these datasets for sequence classification tasks. Our dataset consists of three main types: bacteria, fungi, and viruses. After the data were downloaded, five types of preprocessing were performed: (1) filtering out the data without taxonomic tags; (2) filtering out all the sequence data of the family containing less than 3 sequence entries; (3) filtering out the mixed mitochondrial sequence data in the bacterial dataset; (4) de-weighting the dataset; and (5) filtering out the sequence data containing ambiguous bases (e.g., N, Y, R, etc.).

More specifically, the three basic types of data are as follows:

- Fungi DNA barcode dataset, which was used in Tian et al. (2018), Sun et al. (2022) at the same time. This dataset was downloaded from the Barcode of Life Data System (BLDS) database, and after preprocessing, it yielded 73,140 non-repetitive sequence data, which belonged to 467 families.
- Bacterial genome sequence dataset, which was used in the article by Sun et al. (2022). This dataset was downloaded from the National Center for Biotechnology Information (NCBI) refseq database, and after preprocessing, 16,373 sequences were obtained, which belong to 178 families.
- Viral genome sequence dataset, which was used in Sun et al. (2022). This dataset was downloaded from the refseq database

of the National Center for Biotechnology Information (NCBI), and after preprocessing, 7382 sequences were obtained, which belong to 83 families.

These sequences vary in length from a few hundred bp (e.g. fungi barcodes) to tens of millions of bp (e.g. bacterial genome sequences), and thus there is a clear requirement for length adaptation of the method itself for sequence classification tasks. Sequences from bacterial genome and viral genome datasets can be downloaded from the NCBI database while sequences from fungi DNA barcodes can be downloaded from the BLDS database; see supplementary tables I-(1-3) for detailed accession numbers.

We also used the spike gene coding sequence dataset of SARS-CoV-2 to demonstrate the numerical non-degeneracy of the components of ACNV. This dataset was downloaded from GISAID database of the five main variants: Alpha (206,520 sequences), Beta (16,383 sequences), Gamma (37,754 sequences), Delta (1,242,249 sequences) and Omicron (1,908,829 sequences), and in total 3,411,735 sequences.

Algorithm for computing ACNV. ACNV does have a slight computational disadvantage over existing natural vector methods, and in general it is not as fast as classical natural vectors and k-mer natural vectors. However, we have been developing a fast algorithm for ACNV and have achieved some results. This may limit the further application of the ACNV method to larger datasets. In this study, we used a fixed-length sliding window algorithm to construct ACNV. Based on this algorithm, we can complete the calculation of a ACNV in only single traversal. The idea behind the fast calculating algorithm is that during the iteration process, for any one pair of adjacent subsequences, the calculated value of the previous subsequence is used to continue calculating the next one, thereby decreasing computational complexity.

Specifically, for each pair of adjacent subsequences, the only difference between them is the single characters on the left and right sides, so it is not difficult to find that the calculating of adjacent subsequences have a high degree of repeatability. Therefore, According to the decomposition of formulas mentioned above, we sequentially traverse and calculate the subsequences. When we calculate the next subsequence, we remove the left character of the previous subsequence and add the adjacent right character. Finally, the sum of all same-subsequence values is divided by the number of relevant nucleic acids to complete the calculation.

We could assume that the average length of the nucleic acids sequence is L, and we need to calculate a k-mer asymmetric natural vector $k, L \in \mathbb{N}_+$, where $L \geq k$ for each sequence. The computational complexity of using brute force algorithm for direct enumeration is $O(kL)$, while the computational complexity of this fast algorithm is only $O(L)$. We will soon present an article on ACNV computation on large-scale genome sequence data based on this fast algorithm.

Algorithm to determine whether two convex hulls intersect. We follow (Sun et al., 2021; Tian et al., 2018) to use a linear programming based method to decide whether two convex hulls intersect with each other. Let $P = \{\boldsymbol{p}_1, \boldsymbol{p}_2, \dots, \boldsymbol{p}_m\}$ and $Q = \{\boldsymbol{q}_1, \boldsymbol{q}_2, \dots, \boldsymbol{q}_n\}$ be two point sets in $\mathbb{R}^k$, then $\mathrm{Conv}(P) \cap \mathrm{Conv}(Q) = \emptyset$ is equivalent to that the following linear programming problem has no feasible solution:

$$\begin{aligned} \min \quad & 0, \\ \text{s.t.} \quad & \sum_{i=1}^{m} \lambda_i \boldsymbol{p}_i = \sum_{j=1}^{n} \mu_j \boldsymbol{q}_j, \\ & \sum_{i=1}^{m} \lambda_i = \sum_{j=1}^{n} \mu_j = 1. \end{aligned} \tag{28}$$

We implemented the above algorithms in python, using the linprog library in scipy.optimize for the linear programming solver (Virtanen et al., 2020).

CRediT authorship contribution statement

Guoqing Hu: Writing – review & editing, Methodology, Formal analysis, Data curation. **Tao Zhou:** Writing – original draft, Formal analysis. **Piyu Zhou:** Writing – review & editing, Validation, Formal analysis. **Stephen Shing-Toung Yau:** Supervision, Methodology, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work is supported by the National Natural Science Foundation of China (NSFC) grant (12171275) and the Tsinghua University Education Foundation fund (042202008). The authors would like to thank the anonymous referees for their valuable suggestions, which significantly improved the quality of the paper.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.gene.2025.149532.

Data availability

I have shared my data through supplementary tables.

References

Blaisdell, B.E., 1986. A measure of the similarity of sets of sequences not requiring sequence alignment. Proc. Natl. Acad. Sci. 83 (14), 5155–5159.
Chan, C.X., Bernard, G., Poirion, O., Hogan, J.M., Ragan, M.A., 2014. Inferring phylogenies of evolving sequences without multiple sequence alignment. Sci. Rep. 4 (1), 6504.
Delibaş, E., 2025. Efficient tf-idf method for alignment-free dna sequence similarity analysis. J. Mol. Graph. Model. 109011.
Deng, M., Yu, C., Liang, Q., He, R.L., Yau, S.S.-T., 2011. A novel method of characterizing genetic sequences: genome space with biological distance and applications. PloS One 6 (3), e17293.
Edgar, R.C., 2004. MUSCLE: multiple sequence alignment with high accuracy and high throughput. Nucleic Acids Res. 32 (5), 1792–1797.
Elmarakeby, H.A., Hwang, J., Arafeh, R., Crowdis, J., Gang, S., Liu, D., AlDubayan, S.H., Salari, K., Kregel, S., Richter others, C., 2021. Biologically informed deep neural network for prostate cancer discovery. Nature 598 (7880), 348–352.
Fan, H., Ives, A.R., Surget-Groba, Y., Cannon, C.H., 2015. An assembly and alignment-free method of phylogeny reconstruction from next-generation sequencing data. BMC Genomics 16, 1–18.
Fumagalli, S.E., Padhiar, N.H., Meyer, D., Katneni, U., Bar, H., DiCuccio, M., Komar, A.A., Kimchi-Sarfaty, C., of, Analysis., 2023. 3.5 million sars-cov-2 sequences reveals unique mutational trends with consistent nucleotide and codon frequencies. Virol. J. 20 (1), 31.
Han, G.-S., Li, Q., Li, Y., 2022. Nucleosome positioning based on dna sequence embedding and deep learning. BMC Genomics 23 (Suppl 1), 301.
Haubold, B., Klötzl, F., Pfaffelhuber, P., 2015. andi: Fast and accurate estimation of evolutionary distances between closely related genomes. Bioinformatics 31 (8), 1169–1175.
Ji, Y., Zhou, Z., Liu, H., Davuluri, R.V., 2021. Dnabert: pre-trained bidirectional encoder representations from transformers model for dna-language in genome. Bioinformatics 37 (15), 2112–2120.
Jiao, X., Pei, S., Sun, Z., Kang, J., Yau, S.S.-T., 2021. Determination of the nucleotide or amino acid composition of genome or protein sequences by using natural vector method and convex hull principle. Fundam. Res. 1 (5), 559–564.
Katoh, K., Misawa, K., i. Kuma, K., Miyata, T., 2002. MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. Nucleic Acids Res. 30 (14), 3059–3066.
Lu, Y.Y., Tang, K., Ren, J., Fuhrman, J.A., Waterman, M.S., Sun, F., 2017. Cafe: a c celerated a lignment-f r e e sequence analysis. Nucleic Acids Res. 45 (W1), W554–W559.

Murray, K.D., Webers, C., Ong, C.S., Borevitz, J., Warthmann, N., 2017. Kwip: The k-mer weighted inner product, a de novo estimator of genetic similarity. PLoS Comput. Biol. 13 (9), e1005727.

Ng, P., 2017. dna2vec: Consistent vector representations of variable-length k-mers. arXiv preprint arXiv:1701.06279.

Price, G.R., 1970. Selection and covariance. Nature 227, 520–0521.

Qi, J., Luo, H., Hao, B., 2004. CVTree: a phylogenetic tree reconstruction tool based on whole genomes. Nucleic Acids Res. 32 (suppl_2), W45–W47.

Ren, R., Yin, C., Yau, S.S.-T., 2022. kmer2vec: A novel method for comparing dna sequences by word2vec embedding. J. Comput. Biol. 29 (9), 1001–1021.

Sarumi, O.A., Hahn, M., Heider, D., 2024. Neuralbeds: Neural embeddings for efficient dna data compression and optimized similarity search. Comput. Struct. Biotechnol. J. 23, 732–741.

Shen, W., Li, Y., 2016. A novel algorithm for detecting multiple covariance and clustering of biological sequences. Sci. Rep. 6 (1), 30425.

Sims, G.E., Jun, S.-R., Wu, G.A., Kim, S.-H., 2009. Alignment-free genome comparison with feature frequency profiles (FFP) and optimal resolutions. Proc. Natl. Acad. Sci. 106 (8), 2677–2682.

Sun, N., Pei, S., He, L., Yin, C., He, R.L., Yau, S.S.-T., 2021. Geometric construction of viral genome space and its applications. Comput. Struct. Biotechnol. J. 19, 4226–4234.

Sun, N., Zhao, X., Yau, S.S.-T., 2022. An efficient numerical representation of genome sequence: natural vector with covariance component. PeerJ 10, e13544.

Thompson, J.D., Higgins, D.G., Gibson, T.J., 1994. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. Nucleic Acids Res. 22 (22), 4673–4680.

Tian, K., Zhao, X., Yau, S.S.-T., 2018. Convex hull analysis of evolutionary and phylogenetic relationships between biological groups. J. Theoret. Biol. 456, 34–40.

van Hilten, A., Kushner, S.A., Kayser, M., Ikram, M.A., Adams, H.H., Klaver, C.C., Niessen, W.J., Roshchupkin, G.V., 2021. Gennet framework: interpretable deep learning for predicting phenotypes from genetic data. Commun. Biology 4 (1), 1094.

van Hilten, A., van Rooij, J., Ikram, M.A., Niessen, W.J., van Meurs, J.B., Roshchupkin, G.V., 2024. Phenotype prediction using biologically interpretable neural networks on multi-cohort multi-omics data. NPJ Syst. Biology Appl. 10 (1), 81.

Vinga, S., Almeida, J., 2003. Alignment-free sequence comparison—a review. Bioinformatics 19 (4), 513–523.

Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S.J., Brett, M., Wilson, J., Millman, K.J., Mayorov, N., Nelson, A.R.J., Jones, E., Kern, R., Larson, E., Carey, C.J., Polat, İ., Feng, Y., Moore, E.W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E.A., Harris, C.R., Archibald, A.M., Ribeiro, A.H., Pedregosa, F., van Mulbregt, P., SciPy 1.0 Contributors, 2020. SciPy, 1.0: Fundamental algorithms for scientific computing in Python. Nature Methods 17, 261–272.

Wang, Y., Tian, K., Yau, S.S.-T., 2019. Protein sequence classification using natural vector and convex hull method. J. Comput. Biol. 26 (4), 315–321.

Wen, J., Chan, R.H., Yau, S.-C., He, R.L., Yau, S.S., 2014. K-mer natural vector and its application to the phylogenetic analysis of genetic sequences. Gene 546 (1), 25–34.

Yi, H., Jin, L., 2013. Co-phylog: an assembly-free phylogenomic approach for closely related organisms. Nucleic Acids Res. 41 (7), e75–e75.

Yu, Z., Yang, Z., Lan, Q., Wang, Y., Huang, F., Cai, Y., 2023. Kmer-node2vec: a fast and efficient method for kmer embedding from the kmer co-occurrence graph, with applications to dna sequences. In: 2023 45th Annual International Conference of the IEEE Engineering in Medicine & Biology Society. EMBC, IEEE, pp. 1–4.

Zahin, T., Abrar, M.H., Jewel, M.R., Tasnim, T., Bayzid, M.S., Rahman, A., 2025. An alignment-free method for phylogeny estimation using maximum likelihood. BMC Bioinformatics 26 (1), 77.

Zhao, R., Pei, S., Yau, S.S.-T., 2020. New genome sequence detection via natural vector convex hull method. IEEE/ ACM Trans. Comput. Biology Bioinform. 19 (3), 1782–1793.

Zhao, X., Tian, K., He, R.L., Yau, S.S.-T., 2019. Convex hull principle for classification and phylogeny of eukaryotic proteins. Genomics 111 (6), 1777–1784.

Zhou, Z., Ji, Y., Li, W., Dutta, P., Davuluri, R., Liu, H., 2023. Dnabert-2: Efficient foundation model and benchmark for multi-species genome. arXiv preprint arXiv:2306.15006.

Zhou, Z., Wu, W., Ho, H., Wang, J., Shi, L., Davuluri, R.V., Wang, Z., Liu, H., 2024. Dnabert-s: Learning species-aware dna embedding with genome foundation models. arXiv.

Zielezinski, A., Girgis, H.Z., Bernard, G., Leimeister, C.-A., Tang, K., Dencker, T., Lau, A.K., Röhling, S., Choi, J.J., Waterman, M.S., et al., 2019. Benchmarking of alignment-free sequence comparison methods. Genome Biol. 20 (1), 1–18.