J Multidisciplinary Scientific Journal

MDPI

Article

Improving ISOMAP Efficiency with RKS: A Comparative Study with t-Distributed Stochastic Neighbor Embedding on Protein Sequences

Sarwan Ali and Murray Patterson *

Department of Computer Science, Georgia State University, Atlanta, GA 30303, USA; sali85@student.gsu.edu
* Correspondence: mpatterson30@gsu.edu

Abstract: Data visualization plays a crucial role in gaining insights from high-dimensional datasets. ISOMAP is a popular algorithm that maps high-dimensional data into a lower-dimensional space while preserving the underlying geometric structure. However, ISOMAP can be computationally expensive, especially for large datasets, due to the computation of the pairwise distances between data points. The motivation behind this study is to improve efficiency by leveraging an approximate method, which is based on random kitchen sinks (RKS). This approach provides a faster way to compute the kernel matrix. Using RKS significantly reduces the computational complexity of ISOMAP while still obtaining a meaningful low-dimensional representation of the data. We compare the performance of the approximate ISOMAP approach using RKS with the traditional t-SNE algorithm. The comparison involves computing the distance matrix using the original high-dimensional data and the low-dimensional data computed from both t-SNE and ISOMAP. The quality of the low-dimensional embeddings is measured using several metrics, including mean squared error (MSE), mean absolute error (MAE), and explained variance score (EVS). Additionally, the runtime of each algorithm is recorded to assess its computational efficiency. The comparison is conducted on a set of protein sequences, used in many bioinformatics tasks. We use three different embedding methods based on *k*-mers, minimizers, and position weight matrix (PWM) to capture various aspects of the underlying structure and the relationships between the protein sequences. By comparing different embeddings and by evaluating the effectiveness of the approximate ISOMAP approach using RKS and comparing it against t-SNE, we provide insights on the efficacy of our proposed approach. Our goal is to retain the quality of the low-dimensional embeddings while improving the computational performance.

Keywords: t-SNE; ISOMAP; data visualization; COVID-19

check for updates

Citation: Ali, S.; Patterson, M. Improving ISOMAP Efficiency with RKS: A Comparative Study with t-Distributed Stochastic Neighbor Embedding on Protein Sequences. *J* **2023**, *6*, 579–591. https://doi.org/10.3390/j6040038

Academic Editor: James David Adams

Received: 2 July 2023
Revised: 27 October 2023
Accepted: 29 October 2023
Published: 31 October 2023

1. Introduction

Data visualization is a fundamental tool for exploring and understanding complex datasets [1,2]. High-dimensional data, which often arise in various fields such as bioinformatics, genomics, and image processing, pose significant challenges for visualization [3–5]. ISOMAP is a nonlinear dimensionality reduction technique, a widely used algorithm, that offers a solution by mapping high-dimensional data into a lower-dimensional space while preserving the intrinsic geometric structure of the data. However, it can be computationally and memory-intensive to calculate geodesic distances for huge datasets. When working with extremely high-dimensional data, ISOMAP might still succumb to the curse of dimensionality, similar to many other dimensionality reduction approaches [6]. To solve such problems, we can use an approximate method to reduce the computational and memory costs. For this purpose, we propose a random kitchen sinks (RKS) [7–9] approximation-based distance computation method in this paper. By utilizing RKS in the ISOMAP algorithm, we aim to significantly reduce the computational complexity of the traditional

ISOMAP method while obtaining a meaningful low-dimensional data representation. This approach has the potential to make ISOMAP more scalable and practical for analyzing large datasets.

The motivation behind this research is to improve the efficiency of the ISOMAP algorithm by leveraging an approximate method based on random kitchen sinks (RKS) [7,8]. RKS is a technique that allows for faster computation of the kernel matrix, which can be used to approximate the pairwise distances between data points. It leverages random features to efficiently approximate the geodesic distances without explicitly computing them. For each data point, ISOMAP identifies its k-nearest neighbors based on a distance metric and constructs a nearest-neighbor graph. The local structure of the data is efficiently captured by this graph, which joins nearby data points. The geodesic distances between the data points in the closest neighbor graph are calculated [10]. This critical phase accounts for the curvature and nonlinearity of the underlying manifold. The pairwise geodesic distances are calculated [11], and then the data are projected onto a lower-dimensional space while adhering as much as possible to the pairwise geodesic distances using classical multidimensional scaling (MDS) [12].

RKS begins by generating a set of random features from a random distribution, like a Gaussian distribution or a Rademacher distribution [9]. RKS uses a kernel approximation method to estimate the pairwise inner products between the mapped data points in the high-dimensional space using these random features [7]. The base functions for the mapping of data points are the random features. The main benefit of RKS is that it can approximate geodesic distances well without explicitly creating the nearest neighbor graph or calculating all pairwise geodesic distances [13]. When compared to more established methods like ISOMAP, this can dramatically lower the computational complexity and memory needs [14], especially for large datasets. RKS is particularly helpful because it reduces the computational burden of calculating geodesic distances in the original high-dimensional space.

To assess the performance of the approximate ISOMAP approach using RKS, we compared it with the widely used t-SNE [15] algorithm. Both ISOMAP and t-SNE are popular techniques for visualizing high-dimensional data, but they differ in their underlying principles and computational characteristics. By comparing the results of ISOMAP with those of RKS and t-SNE, we gained insights into the strengths and limitations of each method and their suitability for different types of datasets. To evaluate the quality of the low-dimensional embeddings generated by ISOMAP with RKS and t-SNE, we employed several metrics: mean squared error (MSE), mean absolute error (MAE), and explained variance score (EVS). These metrics provide quantitative measures of the reconstruction error, comparing the distance matrix D computed from the high-dimensional original data with the approximate distance matrix D_{approx} obtained from the low-dimensional embeddings. Additionally, we measured the runtime of each algorithm to assess its computational efficiency.

The comparison and evaluation were performed on a set of protein sequences, which are used in typical bioinformatics tasks such as characterizing SARS-CoV-2 variants, location of infection, or host specificity. Protein sequences are fundamental to understanding protein structure and function, and they are often represented as high-dimensional data. In this study, we employed different embedding methods based on k-mers [16], minimizers [17], and position weight matrix (PWM) [18] to capture various aspects of the underlying structure and relationships between the protein sequences. This allowed us to explore the effectiveness of the approximate ISOMAP approach using RKS across different representations of the protein sequences. By evaluating the effectiveness of the approximate ISOMAP approach using RKS and comparing it against t-SNE on a protein sequence dataset, this research aims to provide insights into the use of the RKS-based method for approximating ISOMAP. The evaluation focused on the quality of the low-dimensional embeddings and their ability to accurately capture the important characteristics of the protein sequences. The findings of this study contribute to the field of data visualization by helping with choosing the appropriate dimensionality reduction method for bioinformatics and other domains dealing with high-dimensional data.

This paper makes the following key contributions to the field of data visualization and dimensionality reduction:

1. **Proposed Approximate ISOMAP using RKS:** We developed an approximate ISOMAP algorithm using random kitchen sinks (RKS) to improve the computational efficiency of ISOMAP while preserving the quality of the low-dimensional embeddings.
2. **Comparative Analysis with t-SNE:** We conducted a comprehensive comparative analysis between the approximate ISOMAP approach with RKS and the traditional t-SNE algorithm. This comparison provides insights into the relative strengths and weaknesses of these methods in terms of accuracy, computational efficiency, and ability to capture the intrinsic structure of the data.
3. **Evaluation on Protein Sequences:** We evaluated the performance of the approximate ISOMAP approach and t-SNE on a dataset of SARS-CoV-2 spike protein sequences. This evaluation included different embedding methods based on *k*-mers, minimizers, and position weight matrix (PWM), allowing for a comprehensive assessment of the algorithms' effectiveness on diverse representations of the protein sequences.
4. **Assessment Metrics:** We introduced and utilized evaluation metrics, including mean squared error (MSE), mean absolute error (MAE), explained variance score (EVS), and runtime, to measure the quality of the low-dimensional embeddings and compare the performance of ISOMAP with that of RKS and t-SNE.

The remainder of this paper is organized as follows: In Section 2, we provide a detailed background on ISOMAP, t-SNE, and the random kitchen sinks (RKS) technique. Section 3 presents the methodology of the proposed approximate ISOMAP approach using RKS. In Section 4, we describe the experimental setup, including the dataset and the different embedding methods employed. Section 5 presents the results and analysis of the comparative evaluation between approximate ISOMAP and t-SNE, including different performance metrics. Finally, in Section 6, we discuss the findings, limitations, and potential future directions of this research.

2. Related Work

In recent years, various dimensionality reduction techniques have been developed to tackle the challenges posed by high-dimensional data visualization. In this section, we review some of the relevant work in the field, focusing on ISOMAP, t-SNE, and related methods. ISOMAP, introduced by [11], is a widely used algorithm for nonlinear dimensionality reduction. It aims to preserve the geodesic distances between data points in low-dimensional space. ISOMAP constructs a neighborhood graph based on pairwise distances and then uses graph-based algorithms, such as the shortest path algorithm, to compute the geodesic distances. While ISOMAP has been successful in preserving the global structure of the data, its computational complexity grows quadratically with the number of data points, making it inefficient for large-scale datasets.

To address the computational limitations of ISOMAP, several approximate methods have been proposed. For example, authors in [19] introduced an approximation algorithm that leverages graph sparsification techniques to reduce the computational complexity of ISOMAP. They demonstrated that their approach achieves comparable performance to the original ISOMAP algorithm while significantly reducing the computation time. Another approach proposed by [20] utilizes landmark-based approximation, where a small set of landmark points is selected to approximate the pairwise distances in the high-dimensional space. By constructing a low-dimensional embedding based on the landmark distances, the computational complexity of ISOMAP is effectively reduced.

While these approximate methods have produced promising results, they still suffer from certain limitations, such as the loss of accuracy in preserving the global structure or the requirement for additional parameter tuning. In this paper, we propose a novel approach that leverages random kitchen sinks (RKS) to approximate the pairwise distances, aiming to improve the efficiency of ISOMAP while maintaining the quality of the low-dimensional embeddings. Authors [21] proposed the approximate geodesic distance matrix, which implies that ISOMAP can be solved by a kernel eigenvalue problem. In another work [22], the authors proposed upgraded landmark-isometric mapping (UL-Isomap) but their aim was to address the problems of landmark selection for hyperspectral imagery (HSI) classification. Other authors proposed a hybrid approach, quantum kitchen sink (QKS, in [23], which uses quantum circuits to nonlinearly transform classical inputs into features. It is inspired by a technique known as random kitchen sinks, whereby random nonlinear transformation can greatly simplify the optimization of machine learning (ML) tasks [24]. However, QKS is more helpful for complex optimization problems that arise in machine learning or simulating quantum systems rather than tasks like visualization.

t-SNE, introduced by van der Maaten and Hinton [15], is another popular dimensionality reduction algorithm known for its ability to preserve the local structure of the data. Unlike ISOMAP, which focuses on preserving global distances, t-SNE emphasizes the preservation of pairwise similarities between nearby data points. It constructs a probability distribution over pairs of high-dimensional data points and a similar distribution over pairs of their low-dimensional counterparts. The algorithm then minimizes the Kullback–Leibler divergence between the two distributions to obtain the low-dimensional embedding. t-SNE has been widely adopted in various domains, including image analysis, natural language processing, and bioinformatics. Many studies have explored the effectiveness of t-SNE in visualizing high-dimensional biological data, such as gene expression profiles [25], single-cell RNA sequencing data [26], and protein structures [27]. t-SNE has demonstrated its ability to reveal meaningful patterns and clusters in complex biological datasets. While t-SNE is effective in preserving local similarities, it may struggle to capture global structures. Additionally, t-SNE's computational complexity scales quadratically with the number of data points, making it challenging to apply to large-scale datasets.

3. Proposed Approach

In this section, we present the proposed approach for improving the efficiency of the ISOMAP algorithm using random kitchen sinks (RKS) [8]. The RKS technique allows for faster computation of the kernel matrix, which approximates the pairwise distances between data points.

The isometric feature mapping (ISOMAP) algorithm is a classical dimensionality reduction technique that aims to preserve the global geometric structure of high-dimensional data in a lower-dimensional space. The algorithm is based on the concept of geodesic distances, which measure the shortest path between two points along the manifold on which the data lay. In the original ISOMAP approach, the algorithm starts by constructing a neighborhood graph, where each data point is connected to its nearest neighbors. Then, ISOMAP computes the pairwise geodesic distances between all data points using graph-based algorithms such as Dijkstra's algorithm. These pairwise distances are used to construct a low-dimensional embedding of the data using techniques like multidimensional scaling (MDS). MDS is responsible for finding a configuration of points in a lower-dimensional space that best approximates the pairwise geodesic distances observed in the high-dimensional space. By mapping the data points to a lower-dimensional space while preserving their pairwise geodesic distances, ISOMAP can reveal the underlying manifold structure of the data and enable effective visualization of its intrinsic geometry.

3.1. Random Kitchen Sinks (RKS)

The RKS algorithm is employed to compute the kernel matrix approximation. Given an input data matrix X and the desired number of components $n_{\text{components}}$, the RKS algorithm proceeds as given in Algorithm 1.

Remark 1. *Note that we use the approximate distance matrix within the traditional ISOMAP instead of the default distance matrix, which requires more computational time.*

Algorithm 1 Random Kitchen Sinks (RKS)

```
Input: $X$ (input data matrix), $n_{\text{components}}$ (number of components)
Output: $D_{\text{all}}$ (kernel matrix approximation)
function PROCEDURE($X$, $n_{\text{components}}$)
    \* Initialize kernel matrix $D_{\text{all}}$ with shape $(n \times n)$, where $n$ is the number of data points *\
    $D_{all}$ = INITIALIZEKERNELMATRIX(X.shape[0])
    \* Initialize the projection matrix $P$ with shape $(d, n_{\text{components}})$, where $d$ is the number of features in the input data *\
    d = length(X[0,:])                                    ▷ Column Length
    P = INITIALIZEPROJECTIONMATRIX(d, $n_{\text{components}}$)
    for $i$ from 1:$n_{\text{components}}$) do
        \* Generate a random vector $r$ with the same number of elements as the input data *\
        r = GENERATERANDOMVECTOR(d)
        \* Divide random vector $r$ with normalize $r$ and store it as the $i$th column of $P$ *\
        P[:, i] = r / NORM(r)
    end for
    \* Loop over the rows and columns of $D_{all}$ *\
    for $i$ from 1:$n$ do
        for $j$ from 1:$n$ do
        \* Calculate the kernel matrix element using the projection matrix $P$ *\
            $D_{\text{all}}[i, j] = X[i]@P \cdot X[j]@P$         ▷ @ → operator for matrix multiplication
        end for
    end for
    Return $D_{\text{all}}$
end function
```

The RKS algorithm initializes the kernel matrix D_{all} and the projection matrix P. It then generates random vectors, normalizes them, and stores them as columns in P. The algorithm computes each element of D_{all} by taking the dot product between the projected data points $X[i]@P$ and $X[j]@P$.

3.2. Approximate Isomap

To improve the efficiency of the ISOMAP algorithm, we utilize the RKS algorithm to approximate the pairwise distances between data points. The following steps outline the proposed approximate ISOMAP approach:

1. Set the desired number of components $n_{\text{components}}$ and the number of nearest neighbors $n_{\text{neighbors}}$.
2. Compute the kernel matrix approximation D using the RKS algorithm: $D = \text{RKS}(X, n_{\text{components}})$.
3. Perform ISOMAP on the approximated distances:
    (a) Initialize the ISOMAP algorithm with the desired number of components and the number of nearest neighbors: $\text{Isomap}(n_{\text{components}}, n_{\text{neighbors}})$.
    (b) Fit the ISOMAP model on D.
4. Compute the reconstruction error using several different metrics:
    (a) Compute the distance matrix approximation D_{approx} obtained from the ISOMAP model
    (b) Calculate the mean squared error (MSE) between the original distance matrix D and the approximated distance matrix D_{approx}: MSE = mean_squared_error(D, D_{approx}).
    (c) Calculate the mean absolute error (MAE) between D and D_{approx}:

$$MAE = mean\_absolute\_error(D, D_{approx})$$

   (d) Calculate the explained variance score (EVS) between D and D_{approx}: EVS $=$ explained_variance_score(D, D_{approx}).

5. Record the ending time and compute the running time.
6. Display the reconstruction error metrics and the running time.

The proposed approximate ISOMAP approach starts by recording the starting time. The number of components $n_{\text{components}}$ and the number of nearest neighbors $n_{\text{neighbors}}$ are set. The RKS algorithm is then used to compute the kernel matrix approximation D based on the input data matrix X and $n_{\text{components}}$. The ending time is recorded, and the running time is computed.

Next, the ISOMAP algorithm is initialized with the desired number of components and nearest neighbors. The ISOMAP model is fitted to the approximated distances D, and the data are transformed using the fit_transform function. The ending time is recorded again, and the running time is computed.

This proposed approach combines the efficiency of the RKS algorithm for approximating the pairwise distances with the ISOMAP algorithm's ability to preserve the intrinsic structure of the data. By utilizing the approximate ISOMAP approach, we aim to achieve a computationally efficient yet effective dimensionality reduction technique for high-dimensional datasets.

4. Experimental Setup

In this section, we provide details about the dataset used in our experiments and discuss the embedding methods used to convert biological sequences into fixed-dimensional numerical representations. The experiments were conducted on a system with an Intel Core i5 processor running at 2.40 GHz and equipped with 32 GB of memory. The system operated on the Windows operating system. For hyperparameter tuning, we used 5-fold cross-validation. Specifically, we used 5 nearest neighbors used for Spike7k, 6 for the Coronavirus Host, and 5 for the Protein Subcellular dataset. Moreover, we used the Euclidean distance metric for calculating nearest neighbors.

4.1. Dataset Statistics

We used the following datasets for experimentation.

4.1.1. Spike7k Dataset

The Spike7k dataset consists of aligned spike protein sequences obtained from the GISAID database (https://www.gisaid.org/, accessed on 15 July 2023). The dataset comprises a total of 7000 sequences, which represent 22 different lineages of coronaviruses (class labels). Each sequence in the dataset has a length of 1274 amino acids. The distribution of lineages in the Spike7k dataset is presented in Table 1.

Table 1. Spike7k (SARS-CoV-2) dataset statistics for 22 variants. The character '-' means that information is not available. The fourth column shows the total number of mutations in spike (S) region and full-length genome (Gen.) [28].

Lineage	Region	Labels	No. Mutations/Gen.	No. of Sequences
B.1.1.7	U.K. [29]	Alpha	8/17	3369
B.1.617.2	India	Delta	8/17	875
AY.4	India	Delta	-	593
B.1.2	USA	-	-	333
B.1	USA	-	-	292
B.1.177	Spain [30]	-	-	243
P.1	Brazil [31]	Gamma	10/21	194
B.1.1	U.K.		-	163
B.1.429	California	Epsilon	3/5	107
B.1.526	New York [32]	Iota	6/16	104

Table 1. *Cont.*

Lineage	Region	Labels	No. Mutations/Gen.	No. of Sequences
AY.12	India	Delta	-	101
B.1.160	France	-	-	92
B.1.351	South Africa [29]	Beta	9/21	81
B.1.427	California [33]	Epsilon	3/5	65
B.1.1.214	Japan	-	-	64
B.1.1.519	USA	-	-	56
D.2	Australia	-	-	55
B.1.221	The Netherlands	-	-	52
B.1.177.21	Denmark	-	-	47
B.1.258	Germany	-	-	46
B.1.243	USA	-	-	36
R.1	Japan	-	-	32
Total	-	-	-	7000

4.1.2. Coronavirus Host

The Coronavirus Host dataset consists of unaligned spike protein sequences along with information about the genus/subgenus and infected host of the clades within the Coronaviridae family. These data were extracted from ViPR [34] and GISAID. The dataset comprises a total of 5558 spike sequences, corresponding to 21 unique hosts, including Bats, Bovines, Cats, Cattle, Equine, Fish, Humans, Pangolins, Rats, Turtles, Weasels, Birds, Camels, Canis, Dolphins, the Environment, Hedgehogs, Monkeys, Pythons, and Swines. In this dataset, the classification tasks are based on the host names, which serve as the class labels. The maximum, minimum, and average lengths of the sequences in this dataset are 1584, 9, and 1272.36, respectively. Additional statistics for this dataset can be found in Table 2.

Table 2. Dataset Statistics for Coronavirus Host data. The total number of sequences is 5558.

Host Name	No. of Sequences	Host Name	No. of Sequences
Humans	1813	Rats	26
Environment	1034	Pangolins	21
Weasel	994	Hedgehog	15
Swines	558	Dolphin	7
Birds	374	Equine	5
Camels	297	Fish	2
Bats	153	Unknown	2
Cats	123	Python	2
Bovines	88	Monkey	2
Canis	40	Cattle	1
Turtle	1		

4.1.3. Protein Subcellular

The Protein Subcellular dataset [35] comprises unaligned protein sequences annotated with information on 11 distinct subcellular locations, which are used as class labels for classification tasks. The dataset contains a total of 5959 sequences. The subcellular locations, along with their respective counts, are presented in Table 3.

4.2. Baseline

As a baseline comparison, we used the standard t-distributed Stochastic Neighbor Embedding (t-SNE) approach [15]. The t-SNE algorithm is a popular dimensionality reduction technique used for visualizing high-dimensional data. The algorithm aims to preserve the local structure of the data points in e low-dimensional space. In the original t-SNE approach, the algorithm starts by computing pairwise similarities between data points in the high-dimensional space. The similarities are typically calculated using a Gaussian kernel function, where points that are close to each other have higher similarity values. Then, t-SNE constructs a probability distribution over pairs of high-dimensional points and a similar probability distribution over pairs of low-dimensional points in a lower-dimensional space. The algorithm iteratively minimizes the divergence between these two probability distributions by adjusting the positions of the low-dimensional points. During each iteration, t-SNE employs a gradient-based optimization technique to update the positions of the low-dimensional points, aiming to better match the pairwise similarities observed in high-dimensional space. By the end of the optimization process, t-SNE generates a low-dimensional representation of the data where nearby points in the high-dimensional space are still close to each other, allowing for effective visualization of the data's structure.

Table 3. Dataset statistics for Protein Subcellular data.

Subcellular Locations	No. of Sequences
Cytoplasm	1411
Plasma Membrane	1238
Extracellular Space	843
Nucleus	837
Mitochondrion	510
Chloroplast	449
Endoplasmic Reticulum	198
Peroxisome	157
Golgi Apparatus	150
Lysosomal	103
Vacuole	63
Total	5959

4.3. Evaluation Metrics

To assess the quality of the low-dimensional embeddings obtained from the approximate ISOMAP approach, reconstruction error metrics were computed. The distance matrix approximation D_{approx} obtained from the ISOMAP model was compared with the original distance matrix D using the mean squared error (MSE), mean absolute error (MAE), and explained variance score (EVS). The ending time was recorded once more, and the final running time was computed.

4.3.1. Mean Squared Error (MSE)

The MSE is defined as follows:

$$MSE = \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{N} (D_{ij} - D_{approx(ij)})^2 \tag{1}$$

where N is the total number of data points, D_{ij} is the pairwise distance between data points i and j in the original high-dimensional space, and $D_{approx(ij)}$ is the pairwise distance between the corresponding low-dimensional embeddings.

4.3.2. Mean Absolute Error (MAE)

The MAE is defined as follows:

$$MAE = \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{N} |D_{ij} - D_{approx(ij)}| \tag{2}$$

where N is the total number of data points, D_{ij} is the pairwise distance between data points i and j in the original high-dimensional space, and $D_{approx(ij)}$ is the pairwise distance between the corresponding low-dimensional embeddings.

4.3.3. Explained Variance Score (EVS)

The EVS is defined as follows:

$$EVS = 1 - \frac{VAR(D - D_{approx})}{Var(D)} \quad (3)$$

where D is the distance matrix computed from the original high-dimensional data, D_{approx} is the distance matrix calculated from the low-dimensional embeddings, $Var(D)$ is the variance of the original distance matrix, and $Var(D - D_{approx})$ is the variance of the difference between the original distance matrix and the distance matrix computed from the low-dimensional embeddings.

4.4. Embedding Generation

To generate the numerical embeddings from the biological sequences, we used the following methods.

4.4.1. Spike2Vec

The Spike2Vec method, as proposed in Ali et al. [36], operates on a protein sequence and produces a spectrum on alphabet Σ, which represents 21 amino acids *ACDEFGHIKLMNPQRSTVWXY* . Specifically, the spectrum comprises a vector for all possible numbers of k-mers, where each bin includes the count/frequency of each k-mer present within a protein sequence. Following alphabetical order, the first bin is for the k-mer AAA, while the last bin of the spectrum is for the k-mer YYY. The total number of bins (i.e., the length of the spectrum) is defined by the following expression:

$$|Spectrum| = |\Sigma|^k \quad (4)$$

This spectrum captures the frequency of occurrence of k-mers within the sequence. In our implementation, we set the value of k to 3, which was determined using a validation set approach. This choice ensured that the spectrum effectively represents the patterns and relationships of trimer subsequences in the protein sequence.

4.4.2. Min2Vec

The Min2Vec embedding method utilizes the concept of minimizers to represent biological sequences in a more concise manner [17]. A minimizer is a modified version of a k-mer, which is a substring of consecutive nucleotides/amino acids, and it is selected as the lexicographically smallest substring (also called m-mer, where $m < k$) in both the forward and backward directions of the k-mer. This approach helps reduce the complexity of representing the sequence.

Formally, for a given k-mer, a minimizer (also known as an m-mer) is defined as a substring of length m that is chosen from the k-mer. The selected substring is the lexicographically smallest in both the forward and backward directions of the k-mer. Here, the values of k and m are chosen empirically, and, for our experiments, we set $k = 9$ and $m = 3$. The choice of these values was determined using a standard validation set approach [37].

Once the minimizers are computed for a given biological sequence, a frequency vector/spectrum-based representation is generated. This representation counts the occurrence of each m-mer (similar to the k-mers spectrum described in Section 4.4.1) within the sequence. The resulting frequency vector captures the distribution of these minimizers and provides a compact representation of the sequence. The length of the m-mers-based spectrum is as follows:

$$|Spectrum| = |\Sigma|^k \quad (5)$$

4.4.3. PWM2Vec

The PWM2Vec [18] approach involves assigning weights to each k-mer within a protein sequence. Specifically, given a sequence, PWM2Vec calculates a position weight matrix (PWM) with dimensions $|\Sigma| \times k$, where Σ represents the 21 amino acid alphabet, and k is the chosen length of the k-mers. The PWM captures the frequency of each amino acid in all k-mers within the sequence. Subsequently, PWM2Vec assigns weights to each k-mer based on their corresponding count values in the PWM. These weights are then utilized as input vectors for machine learning models. In our implementation, we set k to 9 using the standard validation set approach [37].

4.5. Limitation

Since RKS begins by generating a set of random features from a random distribution, the selection of random features could be a challenging task. The distance matrix generated on line 4 of Algorithm 1 can be expensive in terms of memory consumption when the number of data points (i.e., biological sequences) is very large in the input data. For computing reconstruction errors, we used MSE, MAE, and EVS. However, using a more comprehensive list could help us extract more insights regarding the proposed method, which we will explore in future extensions. Moreover, since we obtained the results for datasets comprising only protein sequences, the generalizability of the proposed method could not be fully tested. Hence, it is not clear how the proposed method will behave on other biological datasets, e.g., nucleotides, short-read sequence datasets, etc.

5. Results and Discussion

The results for the approximate ISOMAP and the baseline t-SNE are reported in Table 4 for different embedding methods and datasets. For Spike7k data, the PWM2Vec with ISOMAP outperformed the other methods on all evaluation metrics (other than EVS) and embedding methods including t-SNE. In the case of MSE, the Spike2Vec with ISOMAP showed a 98.7% improvement in performance compared with the Spike2Vec with traditional t-SNE. A similar pattern was observed for MAE, EVS, and computational runtime. For the Coronavirus Host dataset, we again observed that all embedding methods with ISOMAP significantly outperformed the same embedding methods with traditional t-SNE (for all evaluation metrics other than EVS). In the case of MSE, the Spike2Vec with ISOMAP achieved a 99.4% improvement in performance compared with the same Spike2Vec embedding with traditional t-SNE. Similarly, in the case of the Protein Subcellular dataset, we again observed that apart from EVS, all embedding methods with ISOMAP outperformed the same embeddings with t-SNE. The Spike2Vec with ISOMAP achieved an 86.8% improvement in performance compared with the same Spike2Vec with traditional t-SNE. Note that to compute the performance gain, we used the following expression:

$$\% \text{ improvement} \quad = \quad \frac{Val_{t-SNE} - Val_{ISOMAP}}{Val_{t-SNE}} \times 100 \tag{6}$$

where Val_{t-SNE} is the value (i.e., MSE, MAE, EVS, or runtime) computed from t-SNE, while Val_{ISOMAP} is the value computed from ISOMAP.

Table 4. Results comparison for ISOMAP vs. t-SNE using different embedding methods and datasets.

		Spike7k				**Coronavirus Host**				**Protein Subcellular**			
		MSE ↓	**MAE** ↓	**EVS** ↑	**Runtime** ↓	**MSE** ↓	**MAE** ↓	**EVS** ↑	**Runtime** ↓	**MSE** ↓	**MAE** ↓	**EVS** ↑	**Runtime** ↓
t-SNE	Spike2Vec	5389.34	53.19	−0.00	88.56	16,061.80	65.13	0.05	47.38	323.93	13.67	−0.48	82.08
	Min2Vec	4539.75	46.14	−0.03	84.81	11,082.11	59.73	0.12	47.91	2062.37	36.78	−0.11	71.09
	PWM2Vec	5925.47	44.48	−0.05	34.20	91,226.50	221.90	0.09	39.59	4733.79	43.00	−0.04	66.61
ISOMAP	Spike2Vec	66.25	7.17	−22,351.74	81.58	83.43	8.57	−11,597.13	46.56	42.46	6.35	−7032.88	45.79
	Min2Vec	64.26	7.01	−19,921.98	80.32	82.11	8.36	−10,856.24	42.61	62.46	7.76	−7275.28	41.54
	PWM2Vec	55.17	6.62	−8418.65	33.92	81.64	8.52	−6358.49	38.64	55.04	7.26	−6801.26	44.43

Discussion

Since the approximate ISOMAP outperforms the traditional t-SNE in terms of reconstruction error (which means that the ISOMAP might be a more suitable choice when the preservation of global data structure is the priority), we think that the understanding of protein sequences using our approach as an alternative to the t-SNE could help explore the patterns/clusters in the original data. These patterns may be hidden/unexplored while using the traditional methods. More specifically, if there is a new coronavirus variant emerging in the given data (i.e., set of protein sequences), the traditional t-SNE method may not highlight a separate cluster of the new variant, which is possible using the proposed method due to its better performance compared with t-SNE.

Embedding methods like Spike2Vec and Min2Vec have sparse information, i.e., most of the bins in the spectrum could have very small/zero numbers. These methods work by transforming complex data into lower-dimensional representations. In this process, some intricate details of the data might be lost or underrepresented, leading to sparse representations. For example, in the context of Spike2Vec and Min2Vec, these methods operate on spectral data, where each bin in the spectrum represents certain features of the data. However, due to the nature of the data, many of these bins might end up with very small or zero values. When using t-SNE on such sparse embeddings, several issues can arise. t-SNE excels at capturing and preserving local and global patterns within data. However, its effectiveness relies on the distribution of data points in the embedded space. Sparse regions in the embeddings can pose challenges. Specifically, t-SNE might not allocate enough attention to these sparse areas, potentially leading to the omission of crucial patterns, outliers, or rare data instances. In essence, the sparsity in embeddings can cause t-SNE to overlook important data characteristics, resulting in an incomplete representation. In contrast, ISOMAP offers a different approach. Instead of focusing on the values of data points, ISOMAP prioritizes the relationships and distances between data points. It constructs the pairwise distances between data points and then maps these points to a lower-dimensional space while preserving these distances as accurately as possible. By maintaining the relative distances between data points, ISOMAP can capture the overall structure and relationships within the data, even if certain regions are sparse. It does not risk overlooking important patterns that might be masked by sparsity.

For the computational runtime results in Table 4, since the dimension of the embedding is lowest for the PWM2Vec-based embeddings (i.e., it equals the number of k-mers within a protein sequence), its runtime is the shortest among the three embedding methods. Since the embedding dimensionalities for Spike2Vec and Min2Vec are similar, their computational runtimes are closer to each other.

6. Conclusions

In this paper, we proposed a fast and efficient method for the visualization of high-dimensional data. The proposed method is based on the idea of approximating the traditional ISOMAP method using random kitchen sinks (RKS). We compared the performance of the approximate ISOMAP approach with RKS with that of the traditional t-SNE algorithm and evaluated their effectiveness using three different protein sequence datasets. Our findings, based on different evaluation metrics and embedding methods (including k-mers, minimizers, and position weight matrix), demonstrate that the approximate ISOMAP approach using RKS offers promising results for dimensionality reduction. By leveraging RKS, we were able to reduce the computational complexity of ISOMAP while still obtaining meaningful representations of biological sequences. The comparative analysis with the traditional t-SNE method revealed that the approximate ISOMAP approach performed favorably in terms of capturing the intrinsic structure of the data while providing a faster runtime by approximating the geodesic distances' computation. This suggests that the RKS-based method is versatile and can be applied to various domains dealing with high-dimensional data. The results of the proposed method can guide researchers and practitioners in selecting the most suitable dimensionality reduction method for their specific datasets and applications. Future work can explore further enhancements to the approximate ISOMAP approach using other methods such as quantum kitchen sinks, Nystroem kernel approximation, random Fourier features, etc. Optimizing the hyperparameters or investigating their performance on other types of high-dimensional data could also be areas of future investigation. RKS uses random features, and using some techniques to reduce this randomness could be a potential extension of this work. Additionally, incorporating other evaluation metrics or conducting user studies to assess the visual interpretability of the low-dimensional embeddings can provide a more comprehensive understanding of the proposed approach.

Author Contributions: Conceptualization, S.A. and M.P.; methodology, S.A.; software, S.A.; validation, M.P.; formal analysis, S.A. and M.P.; investigation, S.A. and M.P.; resources, S.A. and M.P.; data curation, S.A.; writing—original draft preparation, S.A. and M.P.; writing—review and editing, S.A. and M.P.; visualization, S.A.; supervision, M.P.; project administration, S.A. and M.P.; funding acquisition, S.A. and M.P. All authors have read and agreed to the published version of the manuscript.

Funding: S. Ali is funded by the Molecular Basis of Disease (MBD) fellowship at Georgia State University. M. Patterson is funded by the Startup Grant at Georgia State University.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data available in a publicly accessible repository at https://gisaid.org/ (accessed on 1 July 2023).

Conflicts of Interest: The authors declare no conflict of interest

References

1. Donalek, C.; Djorgovski, S.G.; Cioc, A.; Wang, A.; Zhang, J.; Lawler, E.; Yeh, S.; Mahabal, A.; Graham, M.; Drake, A.; et al. Immersive and collaborative data visualization using virtual reality platforms. In Proceedings of the 2014 IEEE International Conference on Big Data (Big Data), Washington, DC, USA, 27–30 October 2014 ; pp. 609–614.
2. Protopsaltis, A.; Sarigiannidis, P.; Margounakis, D.; Lytos, A. Data visualization in internet of things: Tools, methodologies, and challenges. In Proceedings of the 15th International Conference on Availability, Reliability and Security, Dublin, Ireland, 25–28 August 2020; pp. 1–11.
3. Donoho, D.L. High-dimensional data analysis: The curses and blessings of dimensionality. *AMS Math Chall. Lect.* **2000**, *1*, 32.
4. Clarke, R.; Ressom, H.W.; Wang, A.; Xuan, J.; Liu, M.C.; Gehan, E.A.; Wang, Y. The properties of high-dimensional data spaces: Implications for exploring gene and protein expression data. *Nat. Rev. Cancer* **2008**, *8*, 37–49. [CrossRef] [PubMed]
5. Geng, X.; Zhan, D.C.; Zhou, Z.H. Supervised nonlinear dimensionality reduction for visualization and classification. *IEEE Trans. Syst. Man Cybern. Part B Cybern.* **2005**, *35*, 1098–1107. [CrossRef] [PubMed]
6. Balasubramanian, M.; Schwartz, E.L. The isomap algorithm and topological stability. *Science* **2002**, *295*, 7. [CrossRef] [PubMed]
7. Karimi, A.H. Exploring New Forms of Random Projections for Prediction and Dimensionality Reduction in Big-Data Regimes. Master's Thesis, University of Waterloo, Waterloo, ON, Canada , 2018.
8. Rahimi, A.; Recht, B. Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning. *Adv. Neural Inf. Process. Syst.* **2008**, *21* .
9. Ghojogh, B.; Ghodsi, A.; Karray, F.; Crowley, M. Johnson-Lindenstrauss lemma, linear and nonlinear random projections, random Fourier features, and random kitchen sinks: Tutorial and survey. *arXiv* **2021**, arXiv:2108.04172.
10. Anowar, F.; Sadaoui, S.; Selim, B. Conceptual and empirical comparison of dimensionality reduction algorithms (pca, kpca, lda, mds, svd, lle, isomap, le, ica, t-sne). *Comput. Sci. Rev.* **2021**, *40*, 100378. [CrossRef]
11. Tenenbaum, J.B.; Silva, V.d.; Langford, J.C. A global geometric framework for nonlinear dimensionality reduction. *Science* **2000**, *290*, 2319–2323. [CrossRef]
12. Cox, T.F.; Cox, M.A. *Multidimensional Scaling*; CRC Press: Boca Raton, FL, USA, 2000.
13. Rojo-Álvarez, J.L.; Martínez-Ramón, M.; Munoz-Mari, J.; Camps-Valls, G. *Digital Signal Processing with Kernel Methods*; John Wiley & Sons: Hoboken, NJ, USA, 2018.
14. Aneesh, C.; Hisham, P.; Kumar, S.; Maya, P.; Soman, K. Variance based offline power disturbance signal classification using support vector machine and random kitchen sink. *Procedia Technol.* **2015**, *21*, 163–170. [CrossRef]
15. Van der Maaten, L.; Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579–2605.

16. Ali, S.; Sahoo, B.; Ullah, N.; Zelikovskiy, A.; Patterson, M.; Khan, I. A k-mer based approach for SARS-CoV-2 variant identification. In Proceedings of the International Symposium on Bioinformatics Research and Applications, Shenzhen, China, 26–28 November 2021; pp. 153–164.
17. Roberts, M.; Haynes, W.; Hunt, B.; Mount, S.; Yorke, J. Reducing storage requirements for biological sequence comparison. *Bioinformatics* **2004**, *20*, 3363–3369. [CrossRef] [PubMed]
18. Ali, S.; Bello, B.; Chourasia, P.; Punathil, R.T.; Zhou, Y.; Patterson, M. PWM2Vec: An Efficient Embedding Approach for Viral Host Specification from Coronavirus Spike Sequences. *Biology* **2022**, *11*, 418. [CrossRef]
19. Talwalkar, A.; Kumar, S.; Morhri, M.; Rowley, H.A. Large scale svd and manifold learning. *J. Mach. Learn. Res.* **2013**, *14*, 3129–3152.
20. Yang, D.; Gao, W.; Li, G.; Yuan, H.; Hou, J.; Kwong, S. Exploiting Manifold Feature Representation for Efficient Classification of 3D Point Clouds. *ACM Trans. Multimed. Comput. Commun. Appl.* **2023**, *19*, 1–21. [CrossRef]
21. Choi, H.; Choi, S. Kernel isomap. *Electron. Lett.* **2004**, *40*, 1612–1613. [CrossRef]
22. Sun, W.; Halevy, A.; Benedetto, J.J.; Czaja, W.; Liu, C.; Wu, H.; Shi, B.; Li, W. UL-Isomap based nonlinear dimensionality reduction for hyperspectral imagery classification. *ISPRS J. Photogramm. Remote Sens.* **2014**, *89*, 25–36. [CrossRef]
23. Wilson, C.; Otterbach, J.; Tezak, N.; Smith, R.; Crooks, G.; da Silva, M. Quantum Kitchen Sinks: An algorithm for machine learning on near-term quantum computers. *arXiv* **2019**, arXiv:1806.08321v2.
24. Noori, M.; Vedaie, S.S.; Singh, I.; Crawford, D.; Oberoi, J.S.; Sanders, B.C.; Zahedinejad, E. Adiabatic quantum kitchen sinks for learning kernels using randomized features. *arXiv* **2019**, arXiv:1909.08131.
25. Stam, R.W.; Schneider, P.; Hagelstein, J.A.; van der Linden, M.H.; Stumpel, D.J.; de Menezes, R.X.; de Lorenzo, P.; Valsecchi, M.G.; Pieters, R. Gene expression profiling–based dissection of MLL translocated and MLL germline acute lymphoblastic leukemia in infants. *Blood J. Am. Soc. Hematol.* **2010**, *115*, 2835–2844. [CrossRef]
26. Luecken, M.D.; Theis, F.J. Current best practices in single-cell RNA-seq analysis: A tutorial. *Mol. Syst. Biol.* **2019**, *15*, e8746. [CrossRef]
27. Spirko-Burns, L.; Devarajan, K. Supervised dimension reduction for large-scale "omics" data with censored survival outcomes under possible non-proportional hazards. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2020**, *18*, 2032–2044. [CrossRef] [PubMed]
28. Ali, S.; Sahoo, B.; Khan, M.A.; Zelikovsky, A.; Khan, I.U.; Patterson, M. Efficient Approximate Kernel Based Spike Sequence Classification. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2022**, 1–12. [CrossRef] [PubMed]
29. Galloway, S.E.; Paul, P.; MacCannell, D.R.; Johansson, M.A.; Brooks, J.T.; MacNeil, A.; Slayton, R.B.; Tong, S.; Silk, B.J.; Armstrong, G.L.; et al. Emergence of SARS-CoV-2 b. 1.1. 7 lineage. *Morb. Mortal. Wkly. Rep.* **2021**, *70*, 95–99. [CrossRef] [PubMed]
30. Hodcroft, E.B.; Zuber, M.; Nadeau, S.; Vaughan, T.G.; Crawford, K.H.D.; Althaus, C.L.; Reichmuth, M.L.; Bowen, J.E.; Walls, A.C.; Corti, D.; et al. Emergence and spread of a SARS-CoV-2 variant through Europe in the summer of 2020. *MedRxiv* 2020, *preprint*. https://dpi.org/10.1101/2020.10.25.20219063.
31. Naveca, F.; Nascimento, V.; Souza, V.; Corado, A.; Nascimento, F.; Silva, G.; Costa, Á.; Duarte, D.; Pessoa, K.; Gonçalves, L.; et al. Phylogenetic relationship of SARS-CoV-2 seq. from Amazonas with emerging Brazilian variants harboring mutations E484K and N501Y in the Spike protein. *Virol. Org.* **2021**, *1* , 1–8.
32. West, A.P., Jr.; Wertheim, J.O.; Wang, J.C.; Vasylyeva, T.I.; Havens, J.L.; Chowdhury, M.A.; Gonzalez, E.; Fang, C.E.; Lonardo, S.S.D.; Hughes, S.; et al. Detection and characterization of the SARS-CoV-2 lineage B. 1.526 in New York. *Nat. Commun.* **2021**, *12*, 4886. [CrossRef]
33. Zhang, W.; Davis, B.D.; Chen, S.S.; Sincuir Martinez, J.M.; Plummer, J.T.; Vail, E. Emergence of a novel SARS-CoV-2 variant in Southern California. *Jama* **2021**, *325*, 1324–1326. [CrossRef]
34. Pickett, B.E.; Sadat, E.L.; Zhang, Y.; Noronha, J.M.; Squires, R.B.; Hunt, V.; Liu, M.; Kumar, S.; Zaremba, S.; Gu, Z.; et al. ViPR: An open bioinformatics database and analysis resource for virology research. *Nucleic Acids Res.* **2012**, *40*, D593–D598. [CrossRef]
35. Protein Subcellular Localization. 2023. Available online: https://www.kaggle.com/datasets/lzyacht/proteinsubcellularlocalization (accessed on 10 January 2023).
36. Ali, S.; Patterson, M. Spike2vec: An efficient and scalable embedding approach for COVID-19 spike sequences. In Proceedings of the 2021 IEEE International Conference on Big Data (IEEE BigData 2021), Virtually, 15–18 December 2021; pp. 1533–1540.
37. Devijver, P.; Kittler, J. *Pattern Recognition: A Statistical Approach*; Prentice-Hall: London, UK, 1982; pp. 1–448.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.