Shape-aware Stochastic Neighbor Embedding for Robust Data Visualisations

Tobias Wängberg[1], Chun-Biu Li[1*] and Joanna Tyrcha[1*]

[1*]Department of Mathematics, Stockholm University, Roslagsvägen 101, Kräftriket, 10691, Stockholm, Sweden.

*Corresponding author(s). E-mail(s): cbli@math.su.se; joanna@math.su.se;

Abstract

The t-distributed Stochastic Neighbour Embedding (t-SNE) algorithm has emerged as one of the leading methods for visualising High Dimensional (HD) data in a wide variety of fields, especially for revealing cluster structure in HD single cell transcriptomics data. However, it is well known that t-SNE is often unable to correctly represent hierarchical relationships between clusters and spurious patterns may arise in the visualisations due to incorrect hyper-parameter settings. In this work we propose to combine t-SNE with shape-aware graph distances to mitigate some of the limitations of the original algorithm. We show the advantage of the graph based algorithm on simulated data sets, where we see a significant improvement in visualizing imbalanced and nonlinear clusters, as well as preservation of hierarchical structure, based on quantitative validation indices. Moreover, we propose a particular hyper-parameter setting, different from previously suggested settings, which we find consistently optimal across all the test cases conducted in this work. Lastly, we demonstrate the superior performance in the visualisations of the MNIST image data set as well as single cell transcriptomics gene expression data.

Keywords: Data visualisation, Dimensionality reduction, Graph distance, Dimensionality reduction validation

1 Introduction

Analysing high dimensional (HD) data is an important challenge in a wide variety of fields. In particular, Dimensionality Reduction (DR) techniques have been increasingly used for visualising high dimensional data by projecting the data onto a low dimensional (LD), usually 2D, space. The aim is to reveal the key hidden structures in the HD data, such as clusters or other geometrical arrangements of the data points. One of the most frequently used methods for this purpose is the t-distributed Stochastic Neighbour Embedding (t-SNE) [1]. The t-SNE is able to create compelling visualisations of data with hundreds of dimensions in fields ranging from image processing [1], speech recognition [2], immuno-profiling of COVID-19 patients [3], etc. One important area of application is cell biology where data is collected on gene expressions in individual cells [4–7]. Cells are often characterised by expressions of thousands of different genes, where the t-SNE has enabled visual analysis of the data in the LD embedding. One of the main successes of t-SNE is its ability to capture discrete patterns even for data with very high dimensions compared with traditional DR methods [1], such as principal component analysis (PCA) [8], locally linear embedding [9], ISOMAP [10] and Laplacian eigenmaps [11], etc. The approach taken by t-SNE is to focus on preserving local structures, usually characterised by Euclidean distances (ED), but not taking into account the global arrangement of points. Despite the merits of t-SNE, there have been drawbacks identified in the literature. Specifically, the t-SNE requires the user to define what is meant by local, this is often difficult to assess in practise and an incorrect notion of local can result in spurious patterns appearing in the LD embedding. Furthermore, global patterns are important in many cases, but they are not guaranteed to be preserved by the t-SNE.

To alleviate these limitations, we propose to incorporate graph-based distances into the framework of t-SNE. The first step of the method is to construct a graph in a data-driven way to represent the HD data by only connecting points in small local neighborhoods. Information about the global structures of the constructed graph can then be captured by shape-aware graph distances (the biharmonic distance in this study). In contrast to conventional distance measures, such as the ED, shape-aware graph distances are able to learn the global shapes of the underlying manifold or structure on which the HD data reside. This has an advantage for dimensionality reduction based on distance preservation. For example, if the underlying manifold is a 1D structure embedded in a 3D Euclidean space, and ED based algorithm such as the t-SNE would have to give up global distances to reduce dimension. A DR technique based on a shape aware distance that respects the curvature of the manifold, on the other hand, can reduce dimension without distorting global structure. We term the t-SNE applied to the shape-aware distances SASNE, short for Shape-Aware Stochastic Neighbour Embedding. The original t-SNE applied to conventional distance measures, e.g., ED, is simply referred as t-SNE hereafter.

More recent methods have also been proposed that claim to solve the shortcomings of t-SNE, in particular the Potential of heat diffusion for affinity-based

transition embedding (PHATE) [12] and the Uniform manilfold approximation (UMAP) [13, 14] methods. In this study we compare SASNE to t-SNE, PHATE and UMAP and show that the competing methods are not consistently able to (i) reveal discrete structure (ii) avoid creating spurious discrete structures, and (iii) preserve global and hierarchical structures as well as SASNE.

In order to confirm the advantages of SASNE compared to t-SNE, PHATE and UMAP, we apply the methods to embed both synthetic and real data sets that demonstrate imbalanced, nonlinear, hierarchical and developmental trajectory structures. The real data sets are, respectively, the MNIST data set of handwritten digits and the gene expressions from cells of the mouse brain. Judging the embedding performance is often done simply by visual inspecting the LD embedding, as in some of the previous works [3–7], where the performance of the method is often judged by the amount of discrete structures appearing in the map. However, although discrete structure is shown in the LD map, this may not reflect the structure of the HD data where transitions between clusters are continuous, and not discrete. To this end the quality of the embedding is scrutinized in terms of quantitative validation methods for clustering (the silhouette indices and plots) and for dimensionality reduction (rank-based methods). It was found that SASNE not only shows significantly improvement in preserving both clustering and hierarchical structures at all scales, but also allows us to fix the hyper-parameter of the method, which is commonly chosen by default [4], in a data-driven way.

The outline of the paper is as follows. The theoretical concepts of the SASNE are introduced in Section 2. These include an overview of the t-SNE method, the motivation and evaluation of graph based distances, and the validation methods used to monitor the quality of clustering and dimensionality reduction in the LD embedding. In Section 3, we demonstrate the superior performances of SASNE in capturing nonlinear and hierarchical structures compared to the original t-SNE and UMAP based on ED, as well as the PHATE based on the potential distance (PD), in terms of both synthetic and real data sets.

2 Shape-aware Stochastic Neighbor Embedding

Overview of t-SNE

The t-SNE [1, 2] is a dimensionality reduction method that takes as input a HD data set X and returns the LD (usually 2D) coordinates Y for the purpose of visualization of data patterns and organizations. The basic idea of the method is to transform the distances between data points in both of the HD and LD spaces into probability distributions. How well the distances are preserved are then quantified in terms of a dissimilarity measure (or cost function), with the Kullback-Liebler divergence commonly used, between the two distributions. Variants of t-SNE [15–17] differ from each other in the probability distributions and the dissimilarity measure used in the methods.

The t-SNE directly takes as inputs the distances between points without the need to know the coordinates of the HD feature space. It proceeds by first converting the HD distances into a probability distribution p_{ij}, usually defined by a Gaussian kernel, over all pairs of points x_i and x_j, such that close points have high probability. A key parameter to be set in t-SNE is the 'perplexity' which corresponds to the effective number of neighbours covered by the Gaussian kernel (See Methods for details). The perplexity therefore controls the variable widths of the Gaussian kernel (or the neighborhood ranges) around different data points in the HD space such that points separated beyond this range are considered as faraway.

Another key idea of t-SNE is the use of long-tailed t-distribution for the probability distribution q_{ij} associated with y_i and y_j in the LD space. As a result of the mismatching of the two distributions p_{ij} and q_{ij} at large distances, faraway points beyond the neighbourhood ranges set by the perplexity in the HD space tend to map to much larger distances in the LD space. This is a special claim of t-SNE to mitigate the crowding problem in dimensionality reduction [1]. Moreover, points within the neighbourhood ranges set by the perplexity in the HD space tend to map to points also close in the LD space. These together amplify and better reveal discrete cluster structures provided that an appropriate value of perplexity is chosen. In practice, a default perplexity value of around 30 is often used with the hope that it defines reasonable neighborhood ranges that match with the spatial extents of clusters in the data.

On the other hand, the Kullback-Leibler divergence, given by $\sum_{ij} p_{ij} \log \frac{p_{ij}}{q_{ij}}$, as the cost function is asymmetric in p_{ij} and q_{ij}. This means that short distances in the HD space with large p_{ij} contribute significantly to the cost function, whereas long distances with small p_{ij} contribute less. Consequently, this asymmetric property of the cost function tends to prevent close points in the HD space from getting separated in the LD space (i.e., extrusions are discouraged). However, it does not prevent distant points in the HD space from being mapped close in the LD space despite the mismatching of p_{ij} and q_{ij} at large distances mentioned above. (i.e., intrusions can occur). The optimisation of t-SNE to find the configuration of points Y that minimizes the cost function are generally performed using gradient-based methods. The mathematical details of t-SNE and its optimisation procedures are given in Methods.

Graph distance motivation

The t-SNE schemes [1, 4] are commonly employed to embed HD data based on, e.g., the Euclidean distance (ED) in the HD space. However, many conventional distance measures in the HD space, such as the ED, Hamming distance for data string comparison [18], negative binomial distance for comparison of gene count vectors in single cell RNA sequencing research [16], etc., are often good distance measures only in small local neighbourhoods that are small compared to the extents of nonlinear structures in the data. For instance, the

ED can only be used locally for data points lying on a hemi-sphere since it fails to capture the curved shape of the underlying manifold when comparing remote points. In other words, conventional distance measures fail to capture the global shape and organization of the data structures. This poses a problem when the common perplexity value of 30, which can connect moderately remote points, is used to produce LD embedding of distinct clusters, e.g., for the MNIST data set [1]. On the other hand, a choice of small perplexity that focuses only on preserving small local structures could result in a LD embedding composed of many small spurious clusters that do not exist in the HD data [19]. Furthermore, global structure and hierarchical organization of clusters are likely lost when a small perplexity is used [4, 19]. <– Part of this paragraph concerns the shortcomings of t-SNE and so it can be shortened a bit (TW: does it need to be shortened? I think it can be as is since it's part of the motivation of using t-SNE together with graph distance)

It is therefore generally difficult to choose an appropriate perplexity that is small enough for the convention distance measures to be useful, but large enough to be able to capture global structures in the HD data. Here we propose to employ the graph distances of the HD data as inputs to t-SNE to resolve the above shortcomings. Graph distances, sometimes called shape-aware distances, that better capture the global nonlinear structures where the HD data reside. As will be shown later in the Results, this naturally leads to a choice of large perplexity value that cannot only mitigate the problem with spurious clusters, but also largely preserve the global and hierarchical structures of the HD data.

Graph construction

The first step in evaluating the graph distances is to construct a graph to represent the HD data, where each node in the graph corresponds to a data point and edges represent the local relationships between points. We define local neighborhoods by only connecting each data point x_i to its k nearest neighbors based on, e.g., the ED. A graph similarity matrix w_{ij} with $i, j = 1, \cdots, n$ between data points x_i and x_j is defined as the inverse of the squared ED, $1/\|x_i - x_j\|^2$. Some studies also use the Gaussian kernel for w_{ij}. The inverse of squared ED is suggested in this study to avoid introducing the Gaussian width as an additional parameter. The similarities of disconnected data points are simply set to zero. With the similarity matrix w_{ij}, the constructed graph can also be viewed as a Markov network with transition probability $w_{ij}/\sum_k w_{ik}$ for a transition from node i to node j.

Different from the perplexity, the parameter k in the graph construction specifies the extent of the local neighourhoods where conventional distance measures, e.g., ED, can be used. We therefore choose a value for k that is as small as possible, just to keep the graph connected, that is, for each point x_i one can reach any other data point x_j using only the local connections. Commonly k is found to be around 5 with this method. If the data consists of highly disconnected regions, k may end up being very large to maintain connectivity in the graph. Nevertheless, this case can be handled by first finding the large k for

which the graph is connected, then locating the disconnected components for the $k-1$ nearest neighbor (NN) graph. Locating the disconnected components can be done efficiently in linear time by a depth first search. Keeping the links between the components from the kNN graph, one can then re-run the algorithm recursively on the disconnected components until a lower bound of $k = 5$, say, is reached.

Biharmonic distance

Various graph distances, such as the geodesic distance [20], commute time distance (CTD) [21], diffusion distance [22], etc., exist in defining relationships between nodes that capture the intrinsic geometry of the data. In this study, we employ the biharmonic distance (BHD) [21] to measure distances between points.

Several advantages of employing the BHD are as follows: (i) The BHD between points from the same clusters are usually very small due to the strong within-cluster connectivity in the graph, whereas the BHD between points from different clusters could be very large due to the weak connectivity between clusters. This property of BHD makes discrete structure exaggerated and easier to detect. (ii) Compared with the geodesic distance, the BHD is robust to random noise [21]. (iii) The BHD can be expressed and computed in terms of the eigenvalues and eigenvectors of the graph Laplacian, one of the most fundamental concepts in graph theory [23]. (iv) Unlike, e.g., the diffusion distance, the BHD involves no additional parameter and therefore reduces the subjective input from users. (v) The CTD is similar to the BHD in its computation and points (i)-(iv) holds for the CTD as well. However, a different weighting of the eigenvalues when computing the BHD compared to the CTD leads to a higher stability in estimating large distances [21].

Validation of the low dimensional embedding

In order to monitor the preservation of cluster and hierarchical structures by SASNE, t-SNE, PHATE and UMAP we advocate the use of quantitative validation indices to compare and evaluate the quality of the LD embedding. In previous studies [1, 2], quality of the embedding are often carried out by simple visual inspections but this may lead to misleading conclusions about the data by interpreting spurious patterns created by the methods. To provide a quantitative account of the merits of SASNE compared with the competing methods at the point-wise, cluster-wise (or intermediate), and inter-cluster (or global) scales, we introduce two complementary validation indices, one for clustering and another for dimensionality reduction, as follows.

Cluster validation

In this study, we evaluate how faithfully the embedding preserves the underlying clusters using the silhouette index [24]. For a given point x_i assigned to the cluster C_k ($k = 1, \cdots, K$ with K the number of clusters) containing N_k points, the cohesion a_i is defined as $a_i = \frac{1}{N_k} \sum_{j:j \in C_k} \delta_{ij}$ where δ_{ij} denotes the

distances between points x_i and x_j and the sum runs over all points in the same cluster C_k. Here δ_{ij} is the conventional distance measure, e.g. ED, when the t-SNE or UMAP are used, the BHD when the SASNE is used and the PD when PHATE is used.

To quantify separation, we first define a point-to-cluster distance $\delta(x_i, C_l) = \frac{1}{N_l}\sum_{j:j\in C_l}\delta_{ij}$ where the sum runs over all points in the cluster C_l. For a given point x_i in the cluster C_k, the separation b_i is defined as the distances from x_i to the closest cluster that x_i does not belong to, i.e., $b_i = \min_{l\neq k}\delta(x_i, C_l)$. Combining the cohesion and separation, the point-wise *silhouette value* s_i for point x_i can then be defined as

$$s_i = \frac{b_i - a_i}{\max(a_i, b_i)}. \tag{1}$$

One can see that $-1 \leq s_i \leq 1$ and s_i is close to 1 (-1) for a good (bad) clustering with large (small) separation b_i and small (large) cohesion a_i. Furthermore, the cluster-wise silhouette score $\overline{s}_k$ can be naturally evaluated as the average silhouette value over all points in the cluster C_k,

$$\overline{s}_k = \frac{1}{N_k}\sum_{i:x_i\in C_k} s_i \tag{2}$$

Finally, an overall silhouette coefficient $\overline{S}$ is evaluated by averaging over all clusters,

$$\overline{S} = \frac{1}{K}\sum_{k=1}^{K}\overline{s}_k. \tag{3}$$

We first note that the silhouette index is primarily designed to validate clustering (i.e., unsupervised learning) methods in which the data do not come with labels. Nevertheless, we will apply the silhouette index in Results below to our test and real data sets whose clusters C_k are known, to evaluate how well clustering structures are preserved from the HD space to the LD embedding.

To correctly evaluate clustering results with non-spherical clusters, conventional distance measures, e.g., the ED, which does not contain any shape information, should not be used as the distances δ_{ij} in the silhouette index. Instead, we will show in Results that the use of the BHD is more appropriate. On the other hand, the separation b_i in the silhouette index only considers the closest cluster to the data point under consideration. This means that the silhouette index cannot validate how well hierarchical organizations of clusters at the inter-cluster scales are preserved by the LD embedding. This leads us to introduce a complement validation method that takes the relative placement of the data points into account.

Dimensionality reduction validation

We complement the silhouette index by quantifying how well the relative placement of points in the LD space agrees with those in the HD space. In

dimensionality reduction, preservation of exact distances is too restrictive that can seriously hamper the flexibility of the nonlinear mapping from the HD to the LD space [25]. Instead, it is more desirable for the embedding to only impose a monotonic relationship between the HD and LD distances that corresponds to the preservation of distance rank ordering [26–28]. In addition, unlike classical methods such as PCA and multidimensional scaling, t-SNE and UMAP are not aimed at preserving exact distances.

In this study, a rank-based validation scheme for dimensionality reduction is formulated as follows. For each point x_i in the HD space, the rank vector $\boldsymbol{r_i}^x = (r_{ij}^x)_{j\neq i}$ is defined, where $r_{ij}^x = r$ if x_j is the rth closest point to x_i. The rank vector $\boldsymbol{r}_i^y$ is defined in the same way for the LD space. We then define a point-wise quality measure, $\overline{\boldsymbol{r}}_i$, for the point x_i as the mean absolute rank error (MARE),

$$\overline{\boldsymbol{r}}_i = \frac{1}{n-1} \sum_{j:j\neq i} \frac{\mid r_{ij}^x - r_{ij}^y \mid}{n-1}, \tag{4}$$

to quantify how well the embedding from the HD to LD space preserves the distance ordering relative to the point x_i. Here the MARE is normalised to lie between 0 (perfect rank preservation) and 1 (complete distortion of ranks). Likewise, an overall quality measure of preservation of rank ordering that we term 'average rank error', $\overline{R}$, can simply be evaluated by averaging the point-wise quality over all data points,

$$\overline{R} = \frac{1}{n} \sum_{i=1}^{n} \overline{\boldsymbol{r}}_i. \tag{5}$$

In addition to the quality measures, it is informative to create the rank residual plot (RRP) that allows us to visually inspect the distribution of the rank residuals $r_{ij}^x - r_{ij}^y$. The RRP is a 2D density plot whose ordinate and abscissa are the value of the normalised rank residuals $(r_{ij}^x - r_{ij}^y)/(n-1)$ and the index j $(j = 1, \cdots, n)$, respectively. As we will see in the Results, the RRP also tells us at what scale and to what degree the distance orderings are distorted in the embedding.

3 Results

3.1 Simulation studies

To demonstrate the advantages and provide insights for our graph based approach, we apply the t-SNE, PHATE, UMAP and the SASNE to four simulated test data sets whose clustering structures are known beforehand. These test sets aim to represent different types of data with features that are often found in real data, allowing us to highlight the merits of using graph distances in cluster separation, dimensionality reduction quality and visual clarity.

We show in Fig. 1**a** the first test case of 'imbalanced clusters'. Two clusters are generated from 3D Gaussian distributions with the same variance but with different means and number of points. A good LD (2D) embedding is expected to clearly separate the two clusters.

The second test case where clusters have 'nonlinear structure' is shown in Fig. 1**b**. Each cluster contains 400 points that are sampled along the two underlying 1D curves with Gaussian noise added. A good LD embedding is expected to not only reveal the 1D underlying structures, but also place the data correctly into two distinct groups.

The third test case shown in Fig. 1**c** simulates a data set with 'hierarchical structure', where clusters 1 to 3 and clusters 4 to 6 form two distinct 'super-clusters', respectively. A good LD embedding is expected to reveal this hierarchical structure where the rank ordering of the distances between the six cluster centers is preserved.

In Fig. S1 we show silhouette plots for the above mentioned test cases comparing the BHD, ED and PD where it is informative to see the advantage of using the BHD over both ED and PD in highlighting clustering structures.

The fourth test case, also studied by Moon *et. al.* [12], is illustrated in Fig. 1**d**. This data set contains no discrete structure. Instead, the data contains continuous developmental trajectories which branch off in various directions. This structure is common in single cell data, for example, where cell types continuously differentiate into other kinds of cells. An accurate LD embedding should therefore accurately reveal the different developmental trajectories, and crucially not produce spurious discrete structure. The data set contains 1440 points sampled along linear trajectories in a 60 dimensional space, with added Gaussian noise along each coordinate axis.

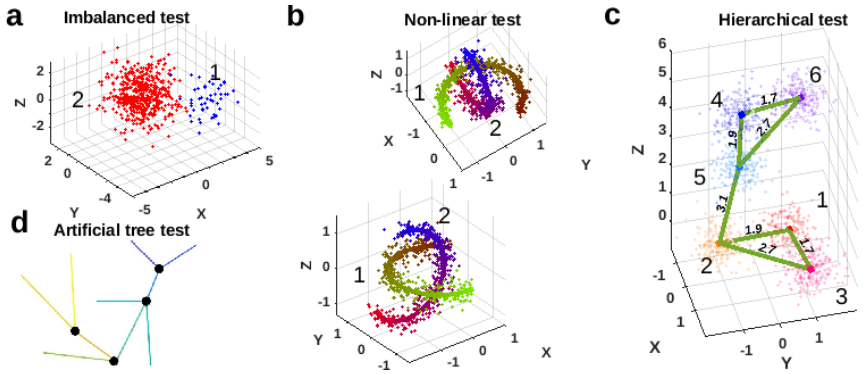

Fig. 1 Three synthetic data sets. **a** Data sampled from two Gaussians with equal covariance matrix but different means. The red and blue clusters contains 1000 and 50 points, respectively. **b** Data sampled uniformly along two non-overlapping 1D nonlinear curves with Gaussian noise added. Each cluster contains 400 points. **c** Data contains 6 clusters with 100 points each. Data are sampled from Gaussians with equal covariance. The cluster means are arranged in two major groups, each containing 3 sub-clusters. The green lines are included for clarity. The numbers next to the lines indicate the ED between the cluster means. **d** Data set containing 1440 points sampled on a piece-wise linear manifold with added Gaussian noise in 60 dimensions.

Choice of hyperparameters

The usual rationale in choosing low perplexity around 30 in the t-SNE is to preserve the local neighborhoods as well as possible. It was also claimed that the t-SNE results are fairy robust against a change of the perplexity value [1]. As Fig. S2 indicate, a low perplexity is in fact a poor choice in terms of the preservation of distance ranks, whereas a large perplexity value, such that the Gaussian kernel also covers remote points, consistently results in significant improvements, especially for the SASNE. We therefore propose a natural choice of perplexity to be around 90 % of the number of points for SASNE, and use this large perplexity value in all the following analyses as a default value. As apparent cluster structures frequently appear in the LD embedding, it may be tempting to choose a lower perplexity when assessing the performance of t-SNE or SASNE qualitatively by eye. However, due to the possible appearance of spurious clusters and the loss of relative placement of clusters, we do not suggest for such choice to avoid making misleading conclusions about the data.

For the remaining methods, we use the default hyperparameter settings (see Methods).

Dimensionality reduction validation

Fig. 2 shows the RRPs and the average rank errors, $\overline{R}$, for the four test cases embedded by the methods. The RRPs show the distance rank preservation at all scales. In particular, distortion of small (large) ranks corresponds to error on the local (global) scale. The local and global scales locate on the left and right sides in the RRP, respectively.

In case of the imbalanced data set (first columns in Fig. 2) , we see that many rank orderings, especially at the intermediate scales located in the middle portion of the abscissa in the RRP, are not accurately preserved. This is expected since three variables are required to describe the relationship between the points from a 3D Gaussian distribution. On the other hand, the SASNE and PHATE show high rank preservation at the very large (inter-cluster level) and small scales are comparable in this test case with simple spherical cluster structures. In contrast, UMAP and t-SNE show poor rank preservation of the larger distances, with slightly better preservation of the local neighborhoods, and with a high average rank error compared to SASNE and PHATE.

From the second column of Fig. 2, we observe for the nonlinear case a significant improvement of the distance rank preservation in the SASNE compared to the other methods, especially t-SNE and UMAP. The RRP shows that the rank ordering at all scales are highly preserved in SASNE (Fig. 2**b**). This is a direct consequence of using the BHD, as a shape-aware distance, that is able to capture the underlying LD nonlinear structure where the data points reside on.

The RRPs of the embeddings of the hierarchical data set by t-SNE, PHATE and UMAP, shown in Fig. 2**g**, **k** and **o** results in mainly the small ranks being preserved while the large ranks are distorted to a higher degree. This means that the hierarchical organization of the clusters is lost in these embeddings. On the other hand, the high preservation of distance ranks by SASNE compared to the other methods is shown in Fig. 2**c**. The main improvement is of due to the preservation of the large distance ranks, meaning that the hierarchical organizations of the clusters are well preserved in the embedding. This is also reflected by the significantly lower average rank error compared to the other methods.

The RRPs evaluating the embeddings of the artificial tree test are shown in the fourth column of Fig. 2. The t-SNE and UMAP preserve mainly the small distance rank, with higher error in terms of preservation of the large distance rank compared to SASNE and PHATE, reflected by the high average rank error. The SASNE achieves the lowest average rank error, with slightly better preservation of the large distance rankings compared to PHATE.

Evaluation of embeddings

From the RRPs in Fig. 2, the preservation of distance ranking is better for SASNE compared to the competing methods in all test cases. This is reflected in the resulting LD embeddings of the imbalanced data shown in the first column of Fig. 3. The SASNE gives very distinct cluster separation that clearly reveals the discrete structure. Indeed, the silhouette coefficient and average silhouette value shown in Fig. 4**a** for the embedding confirm the superior ability of SASNE in highlighting clusters. Furthermore, the cluster separation in the UMAP plot is comparable to that of SASNE, while the t-SNE and PHATE achieves less clear separation of the clusters, where it would be difficult to

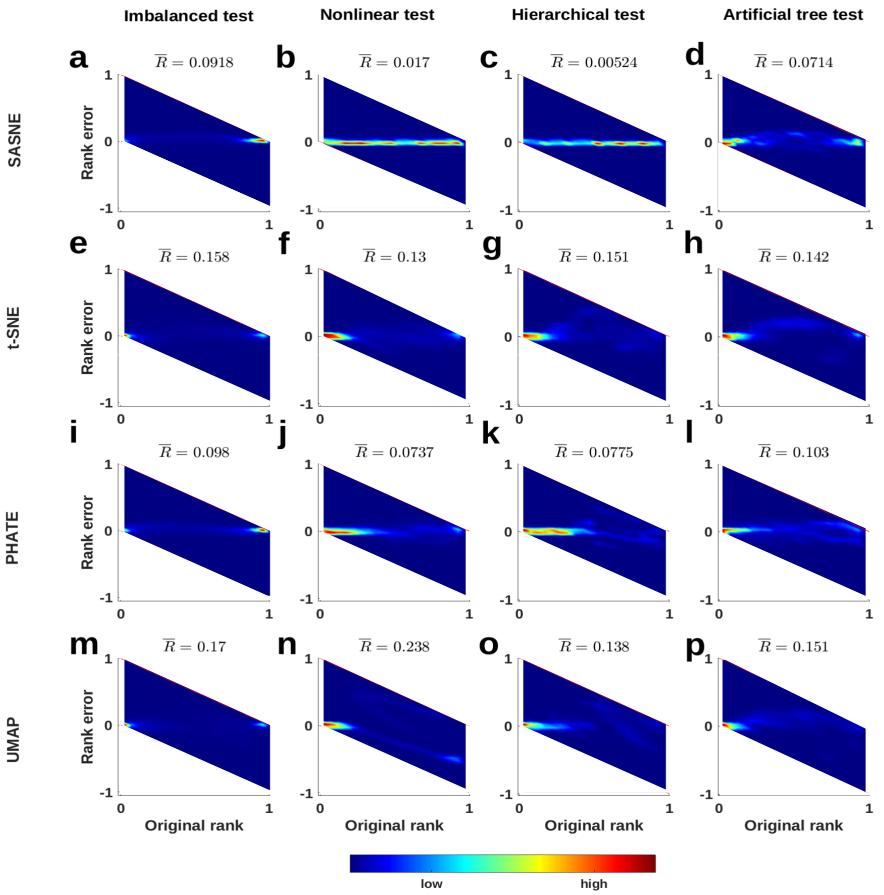

Fig. 2 Rank residual plots (RRP) for the three simulated test cases. The perfect situation in which all distance rank orderings are preserved in the embedding implies that all residuals equal to zero. In that case, RRP shows a shape peak along the horizontal line in the middle of the plot. The residuals are visualised via a 2D histogram, where each bin is colored according to the relative density of points, according to the colormap located at the bottom of the plot. Empty bins are colored white. The red lines indicate the maximum rank distortion. The values on the top right and bottom left of each RRP correspond to the perplexity and the average rank error $\overline{R}$, respectively. The test cases are arranged per column, with the same order as in Fig. 1.

make out the clusters by visual inspection and distinguish it from spurious patterns created by the algorithm.

In case of the nonlinear data set shown in the second column of Fig. 3 one can see the limitation of the t-SNE in that it fragments of one the two clusters into two spurious clusters. This is reflected by the low silhouette score and average silhouette value shown in Fig. 4**b**. The remaining methods successfully untangles the two shapes. However, the SASNE, as shown in Fig. 3**b**, achieves better denoising of the data, thereby clearly revealing the underlying

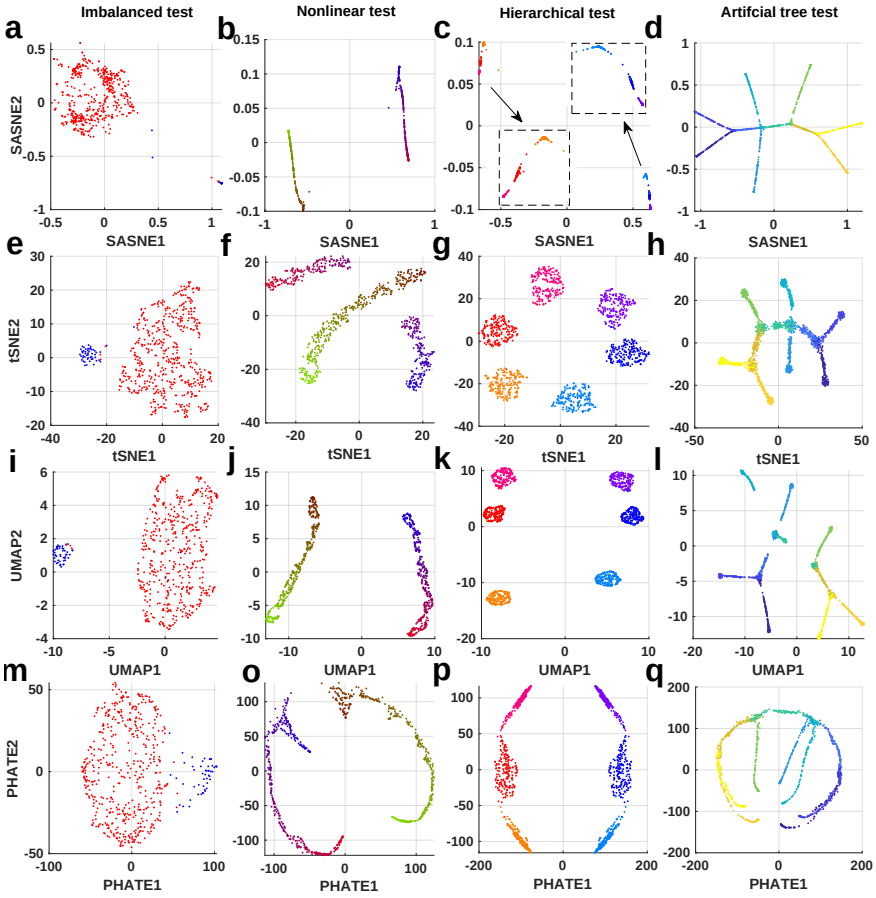

Fig. 3 2D embedding of the test cases in Fig. 1. The color scheme of the clusters are the same as in Fig. 1.

1D structures of the clusters. This improvement in the clustering quality is further confirmed by the silhouette coefficient in Fig. 4**b**.

The t-SNE and SASNE of the hierarchical data set are shown in the third column of Fig. 3. At first glance, the t-SNE and UMAP may be preferred as the spherical shape of the clusters from the original 3D data (Fig. 1**c**) are retained. However, the RRPs in Fig. 2**g** and **o** show that both t-SNE and UMAP introduces distortion at all scales but the very local that cannot be easily detected by eye. On the other hand, the SASNE achieves much lower distance rank distortion at all scales as shown in the right column of Fig. 2**b**, implying that the hierarchical structure of the clusters is well preserved. In terms of cluster validation, the t-SNE correctly separates the individual clusters within each group. The method is however not able to clearly distinguish the reddish group (clusters 1 to 3) from the bluish group (clusters 4-6) as shown in Fig. 3**g**. Moreover UMAP is not able to accurately show the hierarchy as,

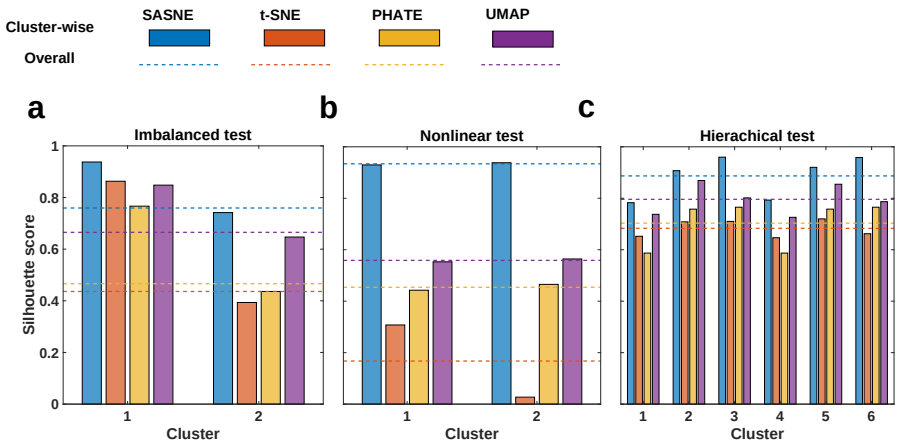

Fig. 4 Barplot showing the average silhouette value for each of the simulated test cases, where clusters are present, and for each method. The dashed lines corresponds to the silhouette coefficient. **a** Barplot showing average silhouette value together with silhouette coefficient for each method for the imbalanced test. **b** Barplot showing average silhouette value together with silhouette coefficient for each method for the nonlinear test. **c** Barplot showing average silhouette value together with silhouette coefficient for each method for the hierarchical test.

for example, cluster 2 and 5 appear equidistant in the embedding which is not an accurate representation of the original data. Similarly, PHATE depicts the two groups symmetrically, which does not reflect the true structure. Contrarily, a better separation of the clusters within each group can be obtained by the SASNE. This improved clustering quality is confirmed by the silhouette coefficient shown in Fig. 4**c** and **f**.

The embeddings of the artificial tree data is found in the fourth column of Fig. 3. The SASNE embedding clearly shows the different branches of the tree while also keeping the trajectories intact. Furthermore SASNE denoises the data, clearly showing the 1D structure of the trajectory. The t-SNE also performs well on this data set, keeping the tree connected, although with less denoising compared to SASNE. Crucially, UMAP shatters the tree, creating many false discrete clusters that do not exist in the data. Although PHATE keeps the tree connected, many some branches are merged together, making them difficult to distinguish by inspection of the visualisation.

In summary, the above test cases demonstrate that the SASNE can reliably embed and reveal clusters with imbalanced, arbitrarily shaped and hierarchical structures based on the qualities of both clustering and preservation of distance ranks, without creating spurious discrete structure that shatters developmental trajectories in the data. Since the shape-aware BHD provides us with a valid global distance measure, the choice of a larger perplexity value, e.g., 90% of the number of points, allows us to consistently fix the only hyper-parameter of the embedding method in a data-driven way. To demonstrate the superior performance of SASNE for real HD data, we consider the following two data sets.

3.2 Gene expression data

We consider a data set of gene expressions from 3663 cells taken from the hippocampal area of a mouse brain [16]. Each cell is characterised by a gene count vector, indicating the expression frequency of the sequenced genes. With the gene count vector as coordinates of the HD space, the data set allows us to identify groupings of cells that correspond to distinct cell types based on their gene expression profiles. In contrast to the simulated data sets and the MNIST data discussed in the next section, the gene expression data is unlabelled, i.e., the corresponding clusters, or cell types, to which the cells belong to are unknown beforehand. Therefore, an additional clustering procedure (not performed here) is needed to group the data points in the LD embedding. Since no cluster label is available, we focus only on how well distance ranks are preserved in the LD embedding and do not consider cluster validation in this case.

Before applying the methods, we follow the same procedures performed by Kobak *et. al.* [4] to reduce the number of features that produce comparable results to those reported by the original works [16] where the data set was obtained. Specifically, we select 1000 representative genes out of 27 998 in total that show high expression levels in a smaller subset of cells, indicating their capability of being good molecular features to distinguish cell types (see Methods). The resulting embeddings of the gene expression data is shown in Fig. 5**a-d**. For comparison, the data is colored according to a previous clustering result performed by Harris *et. al.* [16] that gave rise to a total of 49 clusters by fitting a mixture of binomial distributions using the expectation maximisation algorithm. It has been reported that these cell clusters form hierarchies, where clusters close to each other are indicated by similar colors. Therefore, a better preservation of distance ranks in the LD embedding is important to correctly embed these hierarchies in order to provide meaningful biological interpretations.

From Fig. 5**a**, the SASNE corroborates the previous clustering result that cell groups colored similarly also fall into nearby regions in the SASNE space. For the preservation of hierarchical structures, the RRP shown in Fig. 5**e** confirms a relatively low degree of rank distortion across all scales. Moreover, SASNE shows a pronounced improvement compared to the t-SNE and UMAP according to the RRPs and average rank errors shown in Fig. 5**f** and **h**. Although clustering validation was not performed for this data set, one can still see from Fig. 5**b** and **d** that the t-SNE and UMAP displays better discrete data structures, but with a large distortion of distance ranks across all scales, likely shattering continuous transitions between clusters. Moreover, the PHATE achieves a comparable average rank error, but with the PHATE having higher distortion of the intermediate distance ranks, as can be seen in Fig. 5**e** and **g**. Examining the PHATE embedding in Fig. 5**c** we note the similar shape compared to the artifical tree embedding in Fig. 3**p**, and that some trajectories appear merged by the PHATE. The SASNE, on the other hand, is able to retain both the discrete and hierarchical structures of the data set.

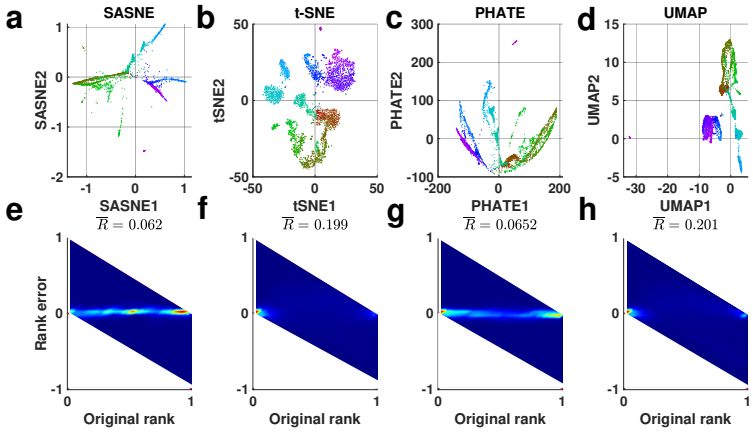

Fig. 5 Results of applying SASNE, t-SNE, PHATE and UMAP to the single cell data set. **a-d** Resulting LD projections by SASNE, t-SNE, PHATE and UMAP respectively. Each data point is colored according to the cluster result of Harris *et. al.* [16]. **e-h** RRPs for each of the LD embeddings.

3.3 MNIST handwritten digit dataset

We now apply the methods to the MNIST data set consisting of gray scale images of handwritten digits [29]. Each image is represented by a 784 (28×28) dimensional vector whose entries correspond to the pixels of the image. The images are labelled based on which digit, from 0 to 9, it corresponds to. This enables us to evaluate how well the images are grouped according to their labels in the LD embedding without the need for extra clustering procedures.

The MNIST data set has known hierarchical structures, for example, digits 4 and 9 look more alike to each other compared to digits 4 and 1. Indeed we confrimed this by examining the overlaps between the digits in Fig. S3. Moreover, from Fig. S3 we see that all digits overlap to some extent with the other digits, only digits 0, 1 and 6 show a clear discrete distinction from the other digits. Hence, many of the digits are not separated into discrete clusters, but have a continuous overlap between each other, which should be reflected in the LD embedding. Some digit clusters are also non-spherically shaped (see Fig. S4) indicating the advantage of using shape-aware distance measures.

Examining the silhouette plots of the distances in the original HD space, found in Fig. S5, we see a low silhouette coefficient for all distance measures, again indicating strong overlap between the clusters. Nevertheless, the BHD are consistent with the conclusions form Fig. S2, confirming the property of increased cluster separation, where digits 0, 1 and 6 are most distinct. The remaining digits does not show clear separation according to the silhouette plot, as found from Fig. S3. On the other hand, the ED shows a silhouette coefficient close to 0, indicating little discrete cluster structure between the digits in the data, except for digit 1 with a higher cluster-wise silhouette value.

Finally, the PD has a comparable silhouette coefficient to the BHD, confirming the advantage of employing graph distances, instead of the ED. However, according to the PD, all digits are roughly equally separated in the HD space, where e.g. digit 1 and 2 have similar silhouette values, which is not consistent with Fig. S3.

We also note that the optimal perplexity for the MNIST data set is again achieved at high perplexity values (see Fig. S7), which is consistent with our conclusion learned from the test cases.

The resulting 2D SASNE is shown in Fig. 6**a**. The embedding shows that digits 0, 1 and 6 form relatively distinct clusters, whereas, for example, digit 2 show overlap between digits 1, 3, 5 and 9, consistent with the plots showing the average 10NN overlaps, found in Fig. S3. Indeed, the RRPs showed in Fig. 6**e-h**, show the significant improvement in preservation of the relative placement of the clusters in the LD embedding by SASNE compared to the other methods, where SASNE show less distortion on all scales. On the other hand, the UMAP plot, although showing clearly separated clusters, disregards the overlaps. Again considering digit 2, in both the UMAP and t-SNE plots, it incorrectly appears completely separated from digit 7. The PHATE more accurately shows the overlap between digits compared to UMAP and t-SNE, but digits that are relatively well separated are not reflected in the PHATE LD map. For example, digit 0 and 6 are merged, and digit 1 is not separated from digit 2 and digit 7.

In terms of the clustering quality, all methods result in a low overall silhouette coefficient (See Fig. 6**i**), as expected due to the small separations, b_i, in the point-wise silhouette value in Eq. (1). Although similar in terms of the silhouette coefficient, the UMAP embedding and t-SNE embedding show most discrete structure. However, according to Fig. S3 and Fig. S5, the observed discrete structure is spurious and does not reflect the true structure of the data as overlap between digits are removed. This is reflected in the SASNE and PHATE embeddings, but not in the t-SNE embedding nor the UMAP embedding. Moreover, PHATE has the lowest cluster separation, and does not reveal the distinct separation of digits 0,1 and 6. In contrast, for digits 0,1 and 6 the cluster-wise silhouette score are higher in the SASNE than in the other methods as seen in Fig. S6. This manifests the ability of the SASNE to amplify true discrete structures in the data, without showing false discrete patterns.

To sum up the analyses of the MNIST data set, the SASNE performs well simultaneously in clustering quality and preservation of distance ranks and hierarchical structure. On the other hand, although UMAP and t-SNE show discrete structure, this is often not an accurate representation of the HD data, where overlap and hierarchical structure is lost in the LD embedding. The PHATE show better preservation of distance ranks compared to t-SNE and UMAP, but is not able to reveal discrete structure in the data as well as the other methods.

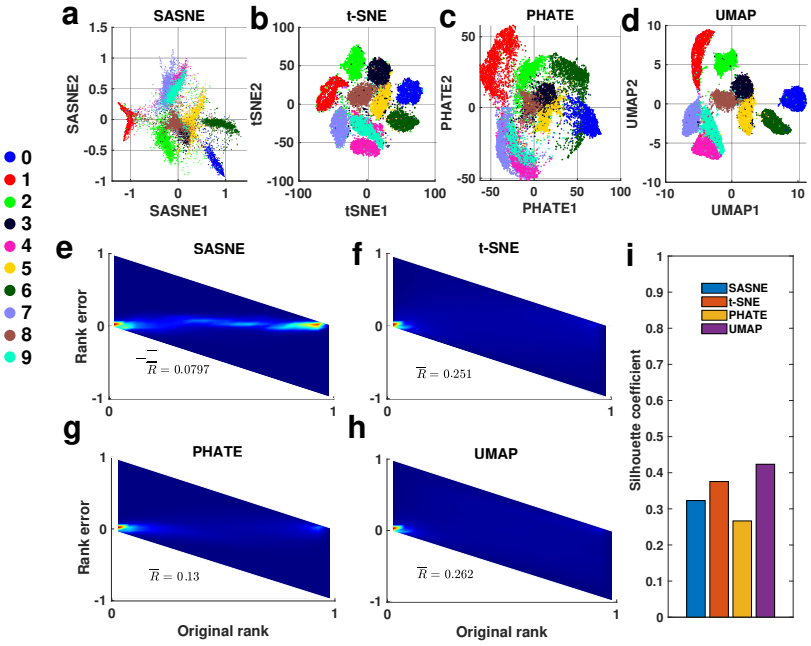

Fig. 6 Results of LD embeddings by SASNE, t-SNE, PHATE and UMAP. **a-d** 2D projections of the MNIST data set using SASNE, t-SNE, PHATE and UMAP respectively. Each point is coloured according to what digit it represents. **e-f** RRPs for the LD embeddings by SASNE, t-SNE, PHATE and UMAP respectively. **i** Barplot showing the silhouette coefficient evaluated on the LD embedding on the MNIST data set for each method.

4 Discussion

By incorporating the concept of shape-aware distances, we proposed in this paper the SASNE and showed how it can mitigate some of the shortcomings of the t-SNE, UMAP and PHATE methods in a data-driven way that can consistently fix the hyper-parameter, perplexity, of the method. In terms of quantitative validation methods in both clustering and dimensionality reduction, the advantages of SASNE in embedding imbalanced, nonlinear and hierarchically structured data, as well as maintaining developmental trajectories without creation of spurious discrete structure with improved denoising, were first demonstrated with simulations where the ground-truth is known. The methods were then applied to two HD real data sets, the single cell gene expression data and the MNIST handwritten digits data set, showing the superior performance of SASNE compared with the current state of the art methods t-SNE, UMAP and PHATE in capturing discrete and hierarchical structures hidden in the HD feature spaces.

It has been claimed in certain cases that the UMAP can outperform t-SNE in computational speed and preservation of global structures [13, 14].

Nevertheless, it was found [4] that the performance of the two methods depends highly on the hyper-parameter settings, and their results could be similar for certain choices of hyper-parameters. The results of the experiments in this work indicate that UMAP in fact distorts the large distance ranks as much, or more, compared to the t-SNE when default parameters are used. The distortion of large distance ranks is to be expected as both t-SNE and UMAP are aimed at preserving local neighborhoods, with a perplexity 30 used for t-SNE and only the 15 NN preserved by default in UMAP, without any estimation of global distances. Moreover, although both t-SNE and UMAP create compelling visualizations with clear discrete structure, they suffer from the creation of spurious discrete structure in the data that can mislead the user.

The PHATE method is similar to SASNE in that it aims at preserving a graph based distance, namely the PD. As our experiments show however, the PD does not amplify the discrete structure to the same extent as the BHD. The PHATE does therefore not reveal as much structure in the embeddings compared to SASNE. Moreover, PHATE relies on the metric multidimensional scaling (MDS) method to embed the PD into the LD space. The MDS has however previously been shown to not perform as well as t-SNE, which is often explained as being due to its inability to handle the crowding problem [1]. Moreover, the PD is defined as the log transformed diffusion distance. The diffusion distance does, however, require the user to decide a parameter t that controls the time-scale of the diffusion. A single time-scale t often cannot capture information of both local and global scales and therefore multiple values of t should be examined to get a complete picture [21, 22]. The authors of PHATE use an elbow method to determine an appropriate t based on the Von Neumann entropy of the normalised eigenspectrum of the graph Laplacian [12]. However, this curve does not contain a clear elbow in general and the choice of t could be quite arbitrary. Lastly, computation of the PD is computationally expensive compared to the BHD, as it requires both diagonalisation and repeated matrix multiplications of the $N \times N$ transition matrix, N being the sample size.

We first note that the computational cost of SASNE may become demanding when the data size n becomes large. In particular, the computation time of the BHD matrix grows as $O(n^2)$ that may cause a problem for data sizes in the tens of thousands. Moreover, the optimization of the t-SNE has computational time of $O(n^2)$. Some numerical approximations have been proposed to speed up the t-SNE optimisation in which the computational time can be reduced to $O(n \log n)$ by tree-based methods [2], and even to $O(n)$ by fast Fourier transform and polynomial interpolation [30]. These approximations do, however, rely on the use of low perplexity values that would sacrifice preservation of global structure. Alternatively, one can downsample the data to a manageable size before applying SASNE. One approach for downsampling can be performed by coarse graining the weighted graph constructed from data in terms of spectral methods [31]. On the other hand, landmarking approaches together with stochastic gradient descent can also be employed to speed up the

t-SNE optimisation [32] and the eigendecompositon of the graph Laplacian in evaluating the BHD matrix [33].

There exist some related studies that also make use of graph-based methods to improve the performance of t-SNE. In particular, Parviainen *et al.* proposed the Graph-SNE (GSNE) method [34] that considers the probability for a random walker to reach data point i from point j and vice versa in a fixed time τ. This probability was then used as the HD distribution p_{ij} in the t-SNE procedures. GSNE has the advantage that speeds up the evaluation of p_{ij} without the need to perform matrix diagonalisation as in the SASNE. However, there is no good strategy in choosing of the hyper-parameter τ that is crucial in determining the 'scale' of the regions explored by the random walker in the graph. Therefore, it was suggested [34] to examine a wide range of diffusion times τ when using GSNE to capture hierarchical structures in the data, which in turn requires several runs of the t-SNE optimizations with increasing computational cost.

Another variant of t-SNE proposed in the literature is the Hierarchical-SNE (HSNE) [32]. The method works similarly to t-SNE but speeds up the computations by a landmarking technique where transition probabilities are computes by Monte Carlo estimation of simulated random walks on the graph representation of the data. The graph is constructed by connecting the 100 nearest neighbors of each data point, where the connections are weighted according to a normalised Gaussian kernel with bandwidth chosen to achieve a perplexity of 100/3. Embeddings at different levels of coarseness can then be created, which would aim to reveal structure at different hierarchies of the data. Although there are computational benefits to the approach, there are hyperparameters that needs to be determined such as the length of the random walks and thresholds for choosing landmarks. Importantly the graph construction connects 100 NN of each data point, where the Euclidean distance may no longer be valid.

~~Furthermore, a recent popular alternative to t-SNE in performing dimensionality reduction of HD data is the Uniform Manifold Approximation and Projection (UMAP) proposed by McInnes *et al.* [13]. It has been claimed in certain cases that the UMAP can outperform t-SNE in computational speed and preservation of global structures [14]. Nevertheless, it was found [4] that the performance of the two methods depends highly on the hyper-parameter settings, and their results could be similar for certain choices of hyper-parameters. On the other hand, the UMAP works similarly [4, 13] to t-SNE by transforming the HD and LD distances to probability distributions based on a defined neighborhood size k similar to the perplexity. To preserve distances with longer ranges in the embedding, the UMAP minimises the cross-entropy, instead of the KL divergence, between the probability distributions. Therefore, it is expected that the shape-aware CTD, as the HD distance, can be readily applied to the UMAP and the idea presented in this study can be employed directly by replacing the t-SNE scheme with that of the UMAP. As a future study, it will be interesting to compare the~~

~~performance of SASNE and UMAP in terms of the quantitative clustering and dimensionality reduction validations.~~

5 Methods

Formalism of t-SNE

Here we provide some mathematical details of the t-SNE method. Suppose there are n data points, the first step is to transform the distances in the HD space into a probability distribution. Specifically, a 'directed' measure of similarity from point x_i to point x_j in the HD space (with $i, j = 1, \ldots, n$) is defined as a conditional probability in terms of the Gaussian kernel and the softmax function,

$$p_{i|j} = \exp\left(-\frac{\delta_{ij}^2}{2\sigma_j^2}\right) \Big/ \sum_{k\neq j} \exp\left(-\frac{\delta_{kj}^2}{2\sigma_j^2}\right), \qquad i \neq j. \tag{6}$$

The self similarity $p_{i|i}$ is set to 0. Here δ_{ij} denotes the distance between points x_i and x_j, which is the conventional distance measure, e.g. ED, in the t-SNE and the BHD in the SASNE. The variable standard deviations σ_j (with $j = 1, \cdots, n$) can be fixed by choosing a constant value for the perplexity, $\mathcal{P}$, defined by

$$\mathcal{P} = 2^{H(p_{\cdot|j})}. \tag{7}$$

In Eq. (7), $H(p_{\cdot|j})$ denotes the Shannon entropy [35] of the probability distribution $p_{\cdot|j}$, defined as $H(p_{\cdot|j}) = -\sum_{i\neq j} p_{i|j} \log p_{i|j}$.

The perplexity can vary between 1 and n and it corresponds to the effective number of neighbors around a point x_j covered by the Gaussian kernel with standard deviation σ_j. Points beyond the perplexity range will simply be counted as 'faraway'. When perplexity equals 1, it corresponds to the case $\sigma_j \to 0$ that all probability mass is placed on the nearest neighbor. On the other hand, when perplexity equals $n-1$, it corresponds to the case $\sigma_j \to \infty$ in which all neighbors are weighted equally. The perplexity is the main hyper-parameter of t-SNE methods that needs to be determined. Moreover, the probability distribution is symmetrised as $p_{ij} = \frac{p_{i|j}+p_{j|i}}{2n}$ for computational convenience.

Similarly, the distances in the LD embedding space are also transformed into a probability distribution in terms of the long-tailed t-distribution with one degree-of-freedom as follows

$$q_{ij} = \frac{(1 + y_i - y_j{}^2)^{-1}}{\sum_{k\neq l}(1 + y_k - y_l{}^2)^{-1}}, \qquad i \neq j. \tag{8}$$

Here in the LD embedding space, the ED is used for both the t-SNE and SASNE. The self-similarity q_{ii} is again set to zero.

The LD embedding coordinates y_i are then obtained by minimizing the Kullback-Leibler (KL) divergence, $\mathrm{KL}(p, q) = \sum_i \sum_{j\neq i} p_{ij} \log \frac{p_{ij}}{q_{ij}}$, as a cost

function between the probability distributions, p_{ij} and q_{ij}, using gradient-based methods. The KL divergence has the property that $\mathrm{KL}(p, q) = 0$ if and only if $p_{ij} = q_{ij}$ for all i and j.

t-SNE optimisation

The optimisation procedures of t-SNE are as follows: In both the t-SNE and SASNE methods, one minimises numerically the KL divergence between the probability distributions, p_{ij} and q_{ij}, as described above by gradient descent. Since the cost function is not convex, the optimisation may converge to a local minimum and therefore the solution may depend on the initialisation, i.e., the initial configuration of the coordinates y_i with $i = 1, \cdots, n$ in the LD space.

In case of optimizing t-SNE, we follow the protocol of Kobak and Berens [4] that the optimisation is initialised with the two leading principal components of the HD data set, normalised by the standard deviation of the corresponding principal component. The initial configuration is further multiplied by a factor of 10^{-4} which was shown empirically to speed up the convergence. For the SASNE optimisation, we use a similar initialisation procedure as in the case of the t-SNE but apply it to the BHD.

We also adopted the optimisation trick to multiply all HD probabilities p_{ij} by a constant $\alpha = 12$, called early exaggeration, for the first 250 iterations. which was shown to lead to better cluster separation [2]. Moreover, as originally suggested by Belkina *et al.* [36], the learning rate in the gradient descent is set to $\eta = n/\alpha$ where n is the number of points which has shown to lead to improved convergence behaviour in terms of stability and speed. Given the above settings, the optimisation was performed by the `tsne` function provided by the `MATLAB` Statistics and Machine Learning Toolbox.

Hyperparameters of PHATE and UMAP

Default paramaters for PHATE are used as suggested in the original publication [12] and for UMAP we follow the default parameters used by Becht *et. al.* [14].

Computing the biharmonic distance

Given a graph G defined by q $n \times n$ similarity matrix W with elements $w_{ij} = 1/\|x_i - x_j\|^2$, one can compute the graph Laplacian $L = D - W$ [23]. Here D is the diagonal degree matrix with elements $d_i = \sum_k w_{ik}$ that is the degree of the node i. The BHD between the points x_i and x_j can be expressed in terms of the eigendecomposition of L as $C_{ij} = \mathrm{Vol}(G) \sum_{k=2}^{n} (v_{ik} - v_{jk})^2 / \lambda_k^2$ [23], where λ_k is the kth eigenvalue, v_{ik} is the ith element of the kth eigenvectors of L, and and $\mathrm{Vol}(G) = \sum_i d_i$ is the volume of the graph G. This expression also shows that the BHD has the form of an ED, i.e., sum of squares $\sum_{k=2}^{n} (z_{ik}^2 - z_{jk}^2)$, with the $(n-1)$D Euclidean coordinates for the ith data point given by $z_{ik} = v_{ik} \sqrt{\mathrm{Vol}(G)/\lambda_k^2}$ $(k = 2, \cdots, n)$. A convenient property of these coordinates are that the corresponding covariance matrix is diagonal. Therefore the PCA initilisation based on the BHD is simply the leading coordinates with largest

corresponding eigenvalues, in this Euclidean space. Hence, after computation of the eigendecomposition of L, these coordinates can be directly input to any standard t-SNE implementation where we keep optimisation scheme consistent with the original t-SNE algorithm.

In this study we computed the BHD using the symmetric Laplacian $L^{\text{sym}} = D^{-\frac{1}{2}} L D^{-\frac{1}{2}}$ [23], instead of L, which empirically produced slightly better results.

Pre-processing of single cell data

We follow the same pre-processing procedures in [4] as follows: Let n_c and n_g be, respectively, the number of cells and the number of genes under consideration. We denote x_{ig} as the expression level of gene g ($g = 1, \cdots, n_g$) in cell i ($i = 1, \cdots, n_c$). The fraction of cells that do not express the gene g is given by $d_g = \frac{1}{n_c} \sum_{i=1}^{n_c} I(x_{ig} = 0)$, where the indicator function $I(x_{ig} = 0) = 1$ when $x_{ig} = 0$, and zero otherwise. Furthermore, the mean log-expression level of the gene g can be expressed as $m_g = \frac{1}{n_{c\neq 0}} \sum_{i:x_{ig}\neq 0} \log x_{ig}$ where $n_{c\neq 0} = \sum_{i=1}^{n_c} I(x_{ij} > 0)$ is the number of cells with non-zero expression of gene g. The next step adopts a heuristic approach from [4] to select 1000 genes by finding a value of b such that there are exactly 1000 genes that exhibit high fraction of zero-expression levels across cells in relation to its mean expression value, which has shown to be able to select biologically relevant genes [37]. Mathematically, this is done by finding a value b such that exactly 1000 genes satisfying the relation $d_g > \exp\left[-\frac{3}{2}(m_g - b)\right] + 0.02$ can be selected. The coefficient $\frac{3}{2}$ and 0.02 are chosen for a good distributional fit [4]. This selected subset of 1000 genes is then kept for the analysis, whereas the others are discarded. Finally, the $\log(1 + x_{ig})$ transformation is applied to the counts of the 1000 selected genes to even out the variance of the larger expression levels. That is, the relative expression difference is considered as opposed to the absolute difference so that, for example, an expression difference from 1 to 5 is considered equal to the difference between 100 and 500.

Code availability

The code will be made available at https://github.com/tobiaswangberg/SASNE.git.

Acknowledgements

We thank Prof. Mats Nilsson and Christoffer Mattsson Langseth for many helpful discussions and comments that helped improve this work. This work is funded by the pair-doctoral program at Stockholm University, titled Statistical methods for spatial tissue profiling.

Author contributions statement

T.W. wrote the manuscript, created the figures and wrote the code for the computer experiments. All authors reviewed the manuscript.

Competing interests

The authors declare no competing interests.

References

[1] van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(11) (2008)

[2] van der Maaten, L.: Accelerating t-sne using tree-based algorithms. The Journal of Machine Learning Research **15**(1), 3221–3245 (2014)

[3] Mathew, D., Giles, J.R., Baxter, A.E., Oldridge, D.A., Greenplate, A.R., Wu, J.E., Alanio, C., Kuri-Cervantes, L., Pampena, M.B., D'Andrea, K., et al.: Deep immune profiling of covid-19 patients reveals distinct immunotypes with therapeutic implications. Science **369**(6508) (2020)

[4] Kobak, D., Berens, P.: The art of using t-sne for single-cell transcriptomics. Nature communications **10**(1), 1–14 (2019)

[5] Scala, F., Kobak, D., Bernabucci, M., Bernaerts, Y., Cadwell, C.R., Castro, J.R., Hartmanis, L., Jiang, X., Laturnus, S., Miranda, E., et al.: Phenotypic variation of transcriptomic cell types in mouse motor cortex. Nature, 1–7 (2020)

[6] Wagner, D.E., Weinreb, C., Collins, Z.M., Briggs, J.A., Megason, S.G., Klein, A.M.: Single-cell mapping of gene expression landscapes and lineage in the zebrafish embryo. Science **360**(6392), 981–987 (2018)

[7] Scala, F., Kobak, D., Shan, S., Bernaerts, Y., Laturnus, S., Cadwell, C.R., Hartmanis, L., Froudarakis, E., Castro, J.R., Tan, Z.H., *et al.*: Layer 4 of mouse neocortex differs in cell types and circuit organization between sensory areas. Nature communications **10**(1), 1–12 (2019)

[8] Pearson, K.: Liii. on lines and planes of closest fit to systems of points in space. The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science **2**(11), 559–572 (1901)

[9] Roweis, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. science **290**(5500), 2323–2326 (2000)

[10] Tenenbaum, J.B., De Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. science **290**(5500), 2319–2323 (2000)

[11] Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. Neural computation **15**(6), 1373–1396 (2003)

[12] Moon, K.R., van Dijk, D., Wang, Z., Gigante, S., Burkhardt, D.B., Chen, W.S., Yim, K., van den Elzen, A., Hirn, M.J., Coifman, R.R., *et al.*: Visualizing structure and transitions in high-dimensional biological data. Nature biotechnology **37**(12), 1482–1492 (2019)

[13] McInnes, L., Healy, J., Melville, J.: Umap: Uniform manifold approximation and projection for dimension reduction. arXiv preprint arXiv:1802.03426 (2018)

[14] Becht, E., McInnes, L., Healy, J., Dutertre, C.-A., Kwok, I.W., Ng, L.G., Ginhoux, F., Newell, E.W.: Dimensionality reduction for visualizing single-cell data using umap. Nature biotechnology **37**(1), 38–44 (2019)

[15] Lee, J.A., Renard, E., Bernard, G., Dupont, P., Verleysen, M.: Type 1 and 2 mixtures of kullback–leibler divergences as cost functions in dimensionality reduction based on similarity preservation. Neurocomputing **112**, 92–108 (2013)

[16] Harris, K.D., Hochgerner, H., Skene, N.G., Magno, L., Katona, L., Gonzales, C.B., Somogyi, P., Kessaris, N., Linnarsson, S., Hjerling-Leffler, J.: Classes and continua of hippocampal ca1 inhibitory neurons revealed by single-cell transcriptomics. PLoS biology **16**(6), 2006387 (2018)

[17] Yang, Z., King, I., Xu, Z., Oja, E.: Heavy-tailed symmetric stochastic neighbor embedding. Advances in neural information processing systems **22**, 2169–2177 (2009)

[18] Waggener, B., Waggener, W.N., Waggener, W.M.: Pulse Code Modulation Techniques. Springer, ??? (1995)

[19] Wattenberg, M., Viégas, F., Johnson, I.: How to use t-sne effectively. Distill (2016). https://doi.org/10.23915/distill.00002

[20] Bouttier, J., Di Francesco, P., Guitter, E.: Geodesic distance in planar graphs. Nuclear physics B **663**(3), 535–567 (2003)

[21] Lipman, Y., Rustamov, R., Funkhouser, T.: Biharmonic distance. ACM Transactions on Graphics **29**(3) (2010)

[22] Coifman, R.R., Lafon, S.: Diffusion maps. Applied and computational harmonic analysis **21**(1), 5–30 (2006)

[23] von Luxburg, U.: A tutorial on spectral clustering. Statistics and computing **17**(4), 395–416 (2007)

[24] Rousseeuw, P.J.: Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of computational and applied mathematics **20**, 53–65 (1987)

[25] Lee, J.A., Verleysen, M.: Nonlinear Dimensionality Reduction. Springer, ??? (2007)

[26] Lee, J.A., Verleysen, M.: Quality assessment of dimensionality reduction: Rank-based criteria. Neurocomputing **72**(7-9), 1431–1443 (2009)

[27] Mokbel, B., Lueks, W., Gisbrecht, A., Hammer, B.: Visualizing the quality of dimensionality reduction. Neurocomputing **112**, 109–123 (2013)

[28] Gracia, A., González, S., Robles, V., Menasalvas, E.: A methodology to compare dimensionality reduction algorithms in terms of loss of quality. Information Sciences **270**, 1–27 (2014)

[29] LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998)

[30] Linderman, G.C., Rachh, M., Hoskins, J.G., Steinerberger, S., Kluger, Y.: Fast interpolation-based t-sne for improved visualization of single-cell rna-seq data. Nature methods **16**(3), 243–245 (2019)

[31] Gfeller, D., De Los Rios, P.: Spectral coarse graining of complex networks. Physical review letters **99**(3), 038701 (2007)

[32] Pezzotti, N., Höllt, T., Lelieveldt, B., Eisemann, E., Vilanova, A.: Hierarchical stochastic neighbor embedding. In: Computer Graphics Forum, vol. 35, pp. 21–30 (2016). Wiley Online Library

[33] Bengio, Y., Paiement, J.-F., Vincent, P., Delalleau, O., Le Roux, N., Ouimet, M.: Out-of-sample extensions for lle, mds, eigenmaps, and spectral clustering. Proceedings of the 16th International Conference on Neural Information Processing Systems, 177–184 (2003)

[34] Parviainen, E., Saramäki: Drawing clustered graphs by preserving neighborhoods. Pattern Recognition Letters **100**, 174–180 (2017)

[35] Shannon, C.E.: A mathematical theory of communication. ACM SIGMOBILE mobile computing and communications review **5**(1), 3–55 (2001)

[36] Belkina, A.C., Ciccolella, C.O., Anno, R., Halpert, R., Spidlen, J., Snyder-Cappione, J.E.: Automated optimized parameters for t-distributed stochastic neighbor embedding improve visualization and analysis of large datasets. Nature communications **10**(1), 1–12 (2019)

[37] Andrews, T.S., Hemberg, M.: M3drop: dropout-based feature selection for scrnaseq. Bioinformatics **35**(16), 2865–2867 (2019)