Looking at Non-Linear Dimension Reductions as Models in the Data Space

Jayani P.G. Lakshika
Econometrics & Business Statistics, Monash University
and
Dianne Cook
Econometrics & Business Statistics, Monash University
and
Paul Harrison
MGBP, BDInstitute, Monash University
and
Michael Lydeamore
Econometrics & Business Statistics, Monash University
and
Thiyanga S. Talagala
Statistics, University of Sri Jayewardenepura

May 21, 2024

Abstract

Nonlinear dimension reduction (NLDR) techniques such as tSNE, and UMAP provide a low-dimensional representation of high-dimensional (high-D) data using non-linear transformation. The methods and parameter choices can create wildly different representations, making it difficult to decide which is best, or whether any or all are accurate or misleading. NLDR often exaggerates random patterns, sometimes due to the samples observed. But NLDR views have an important role in data analysis because, if done well, they provide a concise visual (and conceptual) summary of high-D distributions. To help evaluate the NLDR we have developed an algorithm to show the 2D NLDR model in the high-D space, viewed with a tour. One can see if the model fits everywhere or better in some subspaces, or completely mismatches the data. It is used to evaluate which 2D layout is the best representation of the high-D distribution and see how different methods may have similar summaries or quirks.

Keywords: high-dimensional data, dimension reduction, hexagonal binning, low-dimensional manifold, tour, data vizualization, model in the data space

1 Introduction

Non-linear dimension reduction (NLDR) is popular for making a convenient low-dimensional (k-D) representation of high-dimensional (p-D) data. Recently developed methods include t-distributed stochastic neighbor embedding (tSNE) (van der Maaten & Hinton 2008), uniform manifold approximation and projection (UMAP) (McInnes & Healy 2018), potential of heat-diffusion for affinity-based trajectory embedding (PHATE) algorithm (Moon et al. 2019), large-scale dimensionality reduction Using triplets (TriMAP) (Amid & Warmuth 2022), and pairwise controlled manifold approximation (PaCMAP) (Wang et al. 2021). However, the representation generated can vary dramatically from method to method, and with different choices of parameters or random seeds made using the same method (Figure 1). The dilemma for the analyst is then, **which representation to use**. The choice might result in different procedures used in the downstream analysis, or different inferential conclusions. The research described here provides new visual tools to aid with this decision.

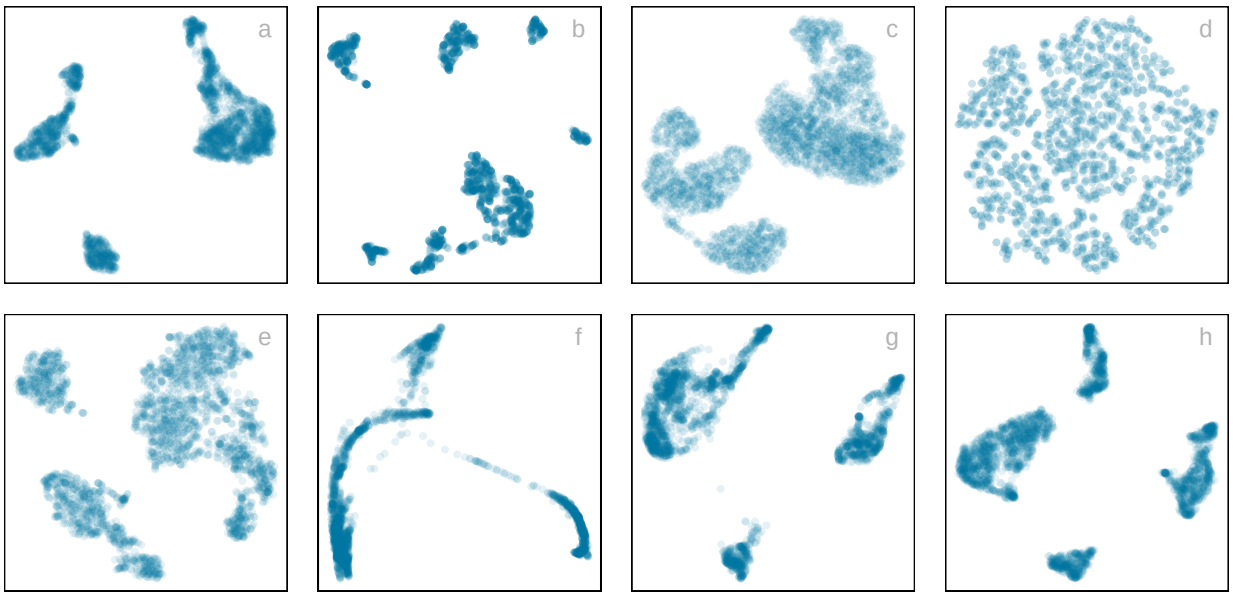


Figure 1: Six different NLDR representations of the same data. Different techniques and different parameter choices are used. Researchers may have seen any of these in their analysis of this data, depending on their choice of method, or typical parameter choice. Would they make different decisions downstream in the analysis depending on which version seen? Which is the most accurate representation of the structure in high dimensions?

The paper is organised as follows. Section 2 provides a summary of the literature on NLDR, and high-dimensional data visualization methods. Section 3 contains the details of the new methodology, including simulated data examples. Two applications illustrating the use of the new methodology for bioinformatics and image classification are in Section 4. Limitations and future directions are provided in Section 5.

2 Background

Historically, k-D representations of p-D data have been computed using multidimensional scaling (MDS) (Borg & Groenen 2005), which includes principal components analysis (PCA) (Jolliffe 2011) as a special case. The k-D representation can be considered to be a layout of points in k-D produced by an embedding procedure that maps the data from p-D. In MDS, the k-D layout is constructed by minimizing a stress function that differences distances between points in p-D with potential distances between points in k-D. Various formulations of the stress function result in non-metric scaling (Saeed et al. 2018) and isomap (Silva & Tenenbaum 2002). Challenges in working with high-dimensional data, including visualization, are outlined in Johnstone & Titterington (2009).

Many new methods for NLDR have emerged in recent years, all designed to better capture specific structures potentially existing in p-D. Here we focus on five currently popular techniques, tSNE, UMAP, PHATE, TriMAP and PaCMAP. tNSE and UMAP can be considered to produce the k-D minimizing the divergence between two distributions, where the distributions are modeling the inter-point distances. PHATE, TriMAP and PaCMAP are examples of diffusion processes (Coifman et al. 2005) spreading to capture geometric shapes, that include both global and local structure.

The array of layouts in Figure 1 illustrate what can emerge from the choices of method and parameters, and the random seed that initiates the computation. Key structures interpreted from these views suggest: (1) highly **separated clusters** (a, b, e, g, h) with the number ranging from 3-6; (2) **stringy branches** (f), and (3) **barely separated clusters** (c, d) which would **contradict** the other representations.

It happens because these methods and parameter choices provide different lenses on the interpoint distances in the data.

The alternative approach to visualizing the high-dimensional data is to use linear projections. PCA is the classical approach, resulting in a set of new variables which are linear combinations of the original variables. Tours, defined by Lee et al. (2021), broaden the scope by providing movies of linear projections, that provide views the data from all directions. Lee et al. (2021) provides an review of the main developments in tours. There are many tour algorithms implemented, with many available in the R package `tourr` (Wickham et al. 2011), and versions enabling better interactivity in `langevitour` (Harrison 2023) and `detourr` (Hart & Wang 2022). Linear projections are a safe way to view high-dimensional data, because they do not warp the space, so they are more faithful representations of the structure. However, linear projections can be cluttered, and global patterns can obscure local structure. The simple activity of projecting data from p-D suffers from piling (Laa et al. 2022), where data concentrates in the center of projections. NLDR is designed to escape these issues, to exaggerate structure so that it can be observed. But as a result NLDR can hallucinate wildly, to suggest patterns that are not actually present in the data.

The solution is to use the tour to examine how the NLDR is warping the space. This approach follows what Wickham et al. (2015) describes as *model-in-the-data-space*. The fitted model should be overlaid on the data, to examine the fit relative the spread of the observations. While this is straightforward, and commonly done when data is 2D, it is also possible in p-D, for many models, when a tour is used.

Wickham et al. (2015) provides several examples of models overlaid on the data in p-D. In hierarchical clustering, a representation of the dendrogrom using points and lines can be constructed by augmenting the data with points marking merging of clusters. Showing the movie of linear projections reveals shows how the algorithm sequentially fitted the cluster model to the data. For linear discriminant analysis or model-based clustering the model can be indicated by $(p-1)$-D ellipses. It is possible to see whether the elliptical shapes appropriately matches the variance of the relevant clusters, and to compare and contrast different fits. For PCA, one can display the k-D plane of the reduced dimension using wireframes of transformed cubes. Using a wireframe is the approach we take here, to represent the NLDR model in p-D.

3 Method

3.1 What is the NLDR model?

At first glance, thinking of NLDR as a modeling technique might seem strange. It is a simplified representation or abstraction of a system, process, or phenomenon in the real world. The p-D observations are the realization of the phenomenon, and the k-D NLDR layout is the simplified representation. From a statistical perspective we can consider the distances between points in the k-D layout to be variance that the model explains, and the (relative) difference with their distances in p-D is the error, or unexplained variance. We can also imagine that the positioning of points in 2D represent the fitted values, that will have some prescribed position in p-D that can be compared with their observed values. This is the conceptual framework underlying the more formal versions of factor analysis (Jöreskog 1969) and multidimensional scaling (MDS) (Borg & Groenen 2005). (Note that, for this thinking the full p-D data needs to be available, not just the interpoint distances.)

We define the NLDR as a function g: $\mathbb{R}^{n\times p} \rightarrow \mathbb{R}^{n\times k}$, with (hyper-)parameters $\boldsymbol{\theta}$. The parameters, $\boldsymbol{\theta}$, depend on the choice of g, and can be considered part of model fitting in the traditional sense. Common choices for g include functions used in tSNE, UMAP, PHATE, TriMAP, PaCMAP, or MDS, although in theory any function that does this mapping is suitable.

With our goal being to make a representation of this 2D layout that can be lifted into high-dimensional space, the layout needs to be augmented to include neighbour information. A simple approach would be to triangulate the points and add edges. A more stable approach is to first hexagonally bin the data, reducing it from n to $m \leq n$ observations, and connect the bin centroids. This process serves to reduce some noisiness in the resulting surface shown in p-D. The steps in this process are shown in Figure 2, and documented below.

To illustrate the method, we use 7-D simulated data, which we call the "S-curve". It is constructed by setting $X_1 = \sin(a), X_2 = U(0,2), X_3 = \text{sign}(a) \times (\cos(a) - 1), \forall a \in [-3\pi/2, 3\pi/2]$. The remaining variables X_4, X_5, X_6, X_7 are all uniform error, with small variance. We would consider $T = (X_1, X_2, X_3)$ to be the true model.

Notation	Description
n, p, k, m	number of observations, variables, embedding dimension, number of non-empty bins, respectively
$\boldsymbol{X}, \boldsymbol{x}$	p-dimensional data (population, sample)
$\boldsymbol{y}$	k-dimensional layout
P	orthonormal basis, generating a *d-dimensional* linear projection of p-dimensional data
T	true model
g	functional mapping from p-D to k-D, especially as prescribed by NLDR
$\boldsymbol{\theta}$	(Hyper-) parameters for NLDR method
r	ranges of the embedding components
$C^{(j)}$	j-dimensional bin centers
(b_1, b_2)	number of bins in each direction
(a_1, a_2)	binwidths, distance between centroids in each direction
$(s_1,\ s_2)$	starting coordinates of the hexagonal grid
q	buffer to ensure hexgrid covers data, proportion of data range, 0-1
b, b'	total and non-empty hexagon bins in the grid
n_m	number of observations within the m^{th} hexagon
h	hexagonal id

Table 1: Summary of notation for describing new methodology.

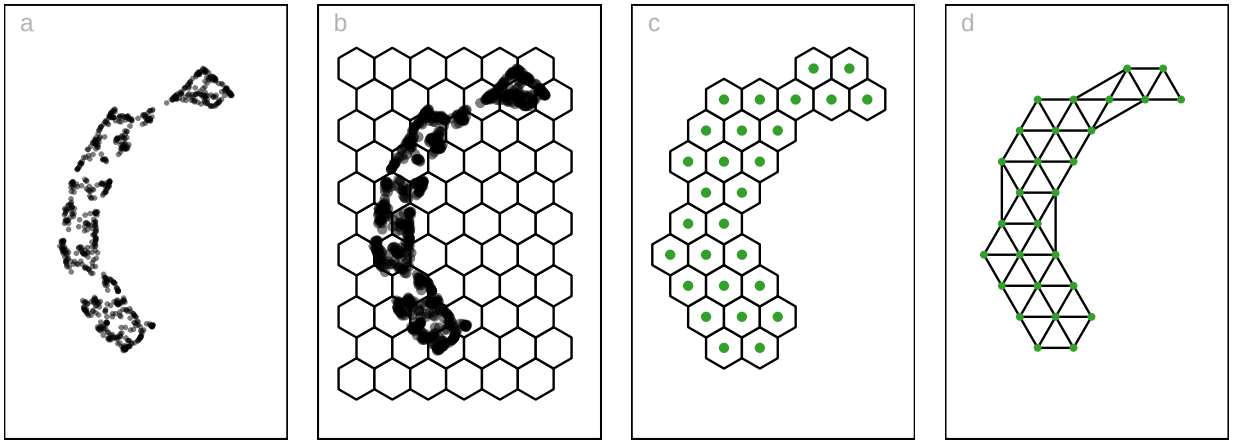


Figure 2: Key steps for constructing the model on the UMAP layout ($k = 2$) of the S-curve data: (a) data, (b) hexagon bins, (c) bin centroids, and (d) triangulated centroids.

3.1.1 Scaling the data

It is beneficial to define the algorithm on data having a standard scale. Here the variables are scaled to [0, 1], but the upper bound can incorporate the aspect ratio produced by the NLDR $(r_1, r_2, ..., r_k)$, by setting them to $(y_{1,\max}, y_{2,\max}, ..., y_{k,\max})$. When $k = 2$ which is assumed for hexagon binning, $y_{1,\max} = 1$ and $y_{2,\max} = \frac{r_2}{r_1}$, as observed in Figure 2.

3.1.2 Computing hexagon grid configurations

The 2-D hexagon grid is defined by the number of bins in each direction (b_1, b_2), giving total number of bins as $b = b_1 \times b_2$, and hexagon id, $h = 1, \ldots, b$. Each hexagon, H_h is uniquely described by centroid, $C_h^{(2)} = (c_{h1}, c_{h2})$. The lower left position where the grid starts at (s_1, s_2), which correspond to the lowest left centroid. The values of s_i need to be below their respective minimum variable values, and could be a full bin lower, to allow a buffer (q) corresponding to a full hexagon width (a_1) and height (a_2) around the data. The values of b_i are variables to be computed that define the reduction in size of the data (n to m).

The value for b_2 is computed by fixing b_1. Considering the lower bound of the NLDR, $a_1 > -2q$, and $a_1 > \frac{1+q}{b_1-1}$. Similarly, according to the upper bound of the NLDR, $a_1 > \frac{2r_2(1+q)}{\sqrt{3}(b_2-1)}$, because $a_2 = \frac{\sqrt{(3)}}{2} a_1$ for regular hexagons. Therefore, $b_2 = \left\lceil 1 + \frac{2r_2(b_1-1)}{\sqrt{3}} \right\rceil$.

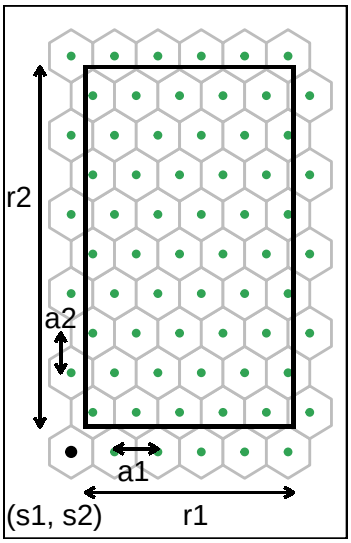


Figure 3: Notations for hexagonal grid configurations.

3.1.3 Binning the data

Points are allocated to the bin they fall into based on the nearest centroid. In situations where a point is equidistant from multiple centroids, tie-breaking rules are applied. If multiple centroids are in the same row, the point is assigned to the leftmost centroid. If multiple centroids are in different rows, the point is assigned to the bottom centroid.

$\{i \in H_h, h = 1, \ldots, b, \text{ and } i = 1, \ldots, n\}$

3.1.4 Summarization of the data in 2D

Let $g : \mathbb{R}^{n\times k} \to \mathbb{R}^{m\times k}$ be the function that maps a point in k-D space to it's bin centroid ($C^{(k)}$). This is the model points in k-D space.

When $k = 2$, one of the representations of a bin is the hexagonal centroid (Figure 2 (c)). Also, it could be the mean of the data points within each hexagon.

3.1.5 Generating edges to indicate neighbors and removing long edges

Delaunay triangulation is used to connect neighboring centroids, which is needed to preserve neighborhood information when the model is lifted into p-D.

When $k = 2$ Delaunay triangulation on $C^{(2)}$ generates the model in 2-D space, which is a triangular mesh (Figure 2 (d)). It generates convex hulls of $C^{(2)}$ such that the circumcircle of every triangle in the triangulation contains no other points from $C^{(2)}$.

When neighboring centroids are connected with edges in Delaunay triangulation in 2-D, there are situations where distant centroids are also connected, which generates long edges. To generate a smooth surface in 2-D, these long edges are removed (Figure 2 (d)).

3.2 Displaying the model in p-D

The last step is to lift the k-D model into p-D by computing p-D vectors that represent bin centroids. Define the set H^k to be the set of points that belong to bin k. We use the p-D Euclidean mean of the points in H^k to map the centroid $C_h^{(k)} = (c_{h1}, c_{h2})$ to a point in p-D. Let the i-th component of the p-D mean be

$$f_i = \frac{1}{|H^k|} \sum_{y\in H^k} y_i,$$

with corresponding p-D vector $\vec{f}_i$. Then, $f(g(x))$ maps a p-D point to the p-D model estimate, completing the model cycle.

Furthermore, edges that exist between k-D representations should also generate edges in p-D by connecting p-D mapping of the corresponding k-D representations.

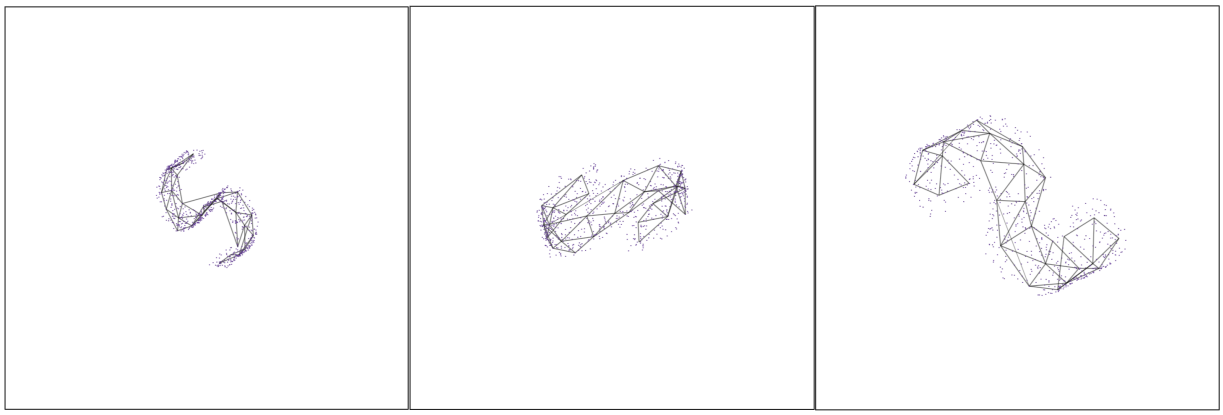


Figure 4: Screen shots of the **langevitour** of the S-curve, shows the model-in-data space, a video of the tour animation is available at (https://youtu.be/G1m4q9k--v4).

3.3 Measuring the fit

3.3.1 Fitted values

The prediction approach involves performing the K-nearest neighbors (KNN) algorithm for an unsupervised classification problem. First, the nearest p-D model point is identified for a given new p-D point. Then, the corresponding k-D centroid mapping for the identified p-D model point is determined. Finally, the coordinates of this k-D centroid are used as the predicted k-D embedding for the new p-D data point. .

3.3.2 Error calculation

To assess how well our method captures and represents the underlying structure of the p-D data, residuals are important. Residuals are computed by taking the squared p-D Euclidean distance, $\sum_{i=1}^{n}\sum_{j=1}^{p}(x_{ij} - C^{(p)}_{x_ij})^2$. Additionally, Mean Squared Error (MSE), $\sum_{i=1}^{n}\sum_{j=1}^{p}\frac{(x_{ij}-C^{(p)}_{x_ij})^2}{n}$, is computed based on residuals.

3.4 Tuning

The performance and robustness of our model depend on three key parameters: (i) the total number of bins (b), (ii) a benchmark value used to remove low-density hexagons, and (iii) a benchmark value used to remove long edges. However, there is no analytical formula to calculate an appropriate value for these parameters. The selection of these parameter values depends on the model performance computed by MSE (see Section 3.3).

3.4.1 Choice of bins

The number of hexagonal bins in the hexagonal grid has a considerable impact on the construction of the 2-D model, serving as the initial step. The chosen total number of bins must effectively capture the structure of the NLDR data. If the number of bins is too low, the model may not be able to capture the structure of the NLDR data effectively (see Figure 5 (a)), while if there are too many bins, it may result in over-fitting the individual

points of the NLDR data (see Figure 5 (c)). Therefore, it is important to determine an effective number of bins to construct a reasonable model (see Figure 5 (b)).

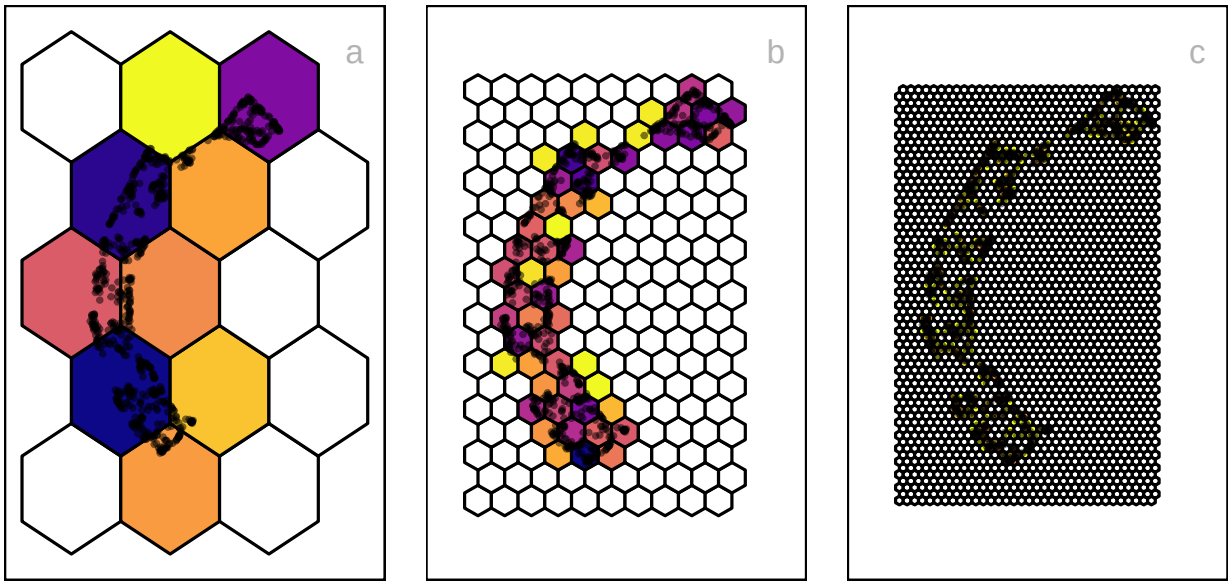


Figure 5: Hexbin plots from different number of bins for the UMAP applied to S-curve data: (a) $b = 12$ (3, 5), (b) $b = 190$ (10, 19), and (c) $b = 2176$ (34, 64). The hexbins are colored based on the density of points, with darker colors indicating higher density and yellow color representing lower density within each bin. What is the number of bins that would be effective?

To determine the effective b, candidate values are selected based on the range between the minimum and approximate maximum b_1, because b_2 is computed from b_1. The minimum b_1 is set to 2, while the maximum number is estimated by taking the square root of $\frac{n}{2}$. The analysis evaluates the MSE across varying b within this range, covering the minimum to maximum values along both axes (see Figure 6).

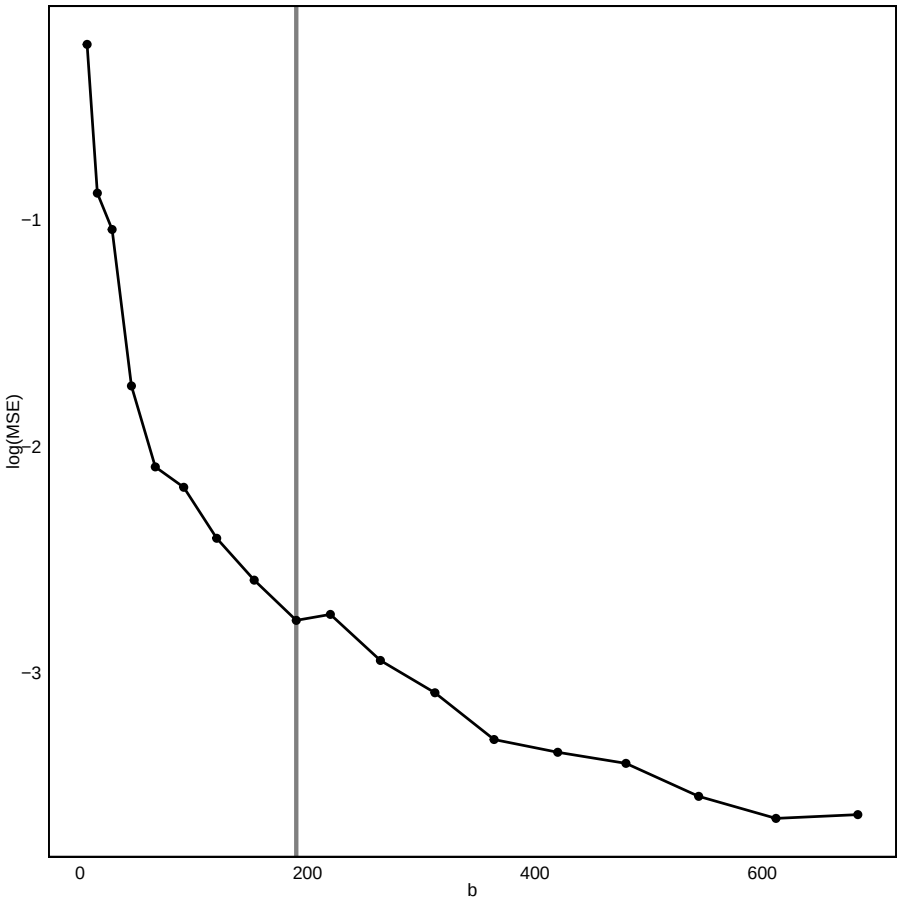


Figure 6: MSE from UMAP applied to S-curve dataset with diffferent b choices. What is the effective b to create the model? The residual plot have a steep slope at the beginning, indicating that a smaller b causes a larger amount of MSE. Then, the slope gradually declines or level off, indicating that a higher b generates a smaller MSE. Using the elbow method, it was observed that when the b is set to 190, the lowest MSE occurred.

3.4.2 Removal of low density bins

Once setting up the hexagonal grid with an appropriate number of bins, some hexagonal bins may have few or no data points within them (see Figure 5 (b)). To ensure comprehensive coverage of the NLDR data, it is necessary to select hexagonal bins with a considerable number of data points. This involves calculating the number of points within each hexagon. Then, the standard count is computed by dividing the number of points within each hexagon by the maximum number of points in the grid. Next, bins with a standard count less than a benchmark value are removed (see Figure 9 (a)). There is no specific rule for selecting a benchmark value. However, the following steps can help determine a suitable value for removing low-density hexagons:

1. Plot the distribution of the standardized counts (see Figure 7).
2. Examine the distribution of counts.
3. Select the first quantile value if the distribution is skewed.

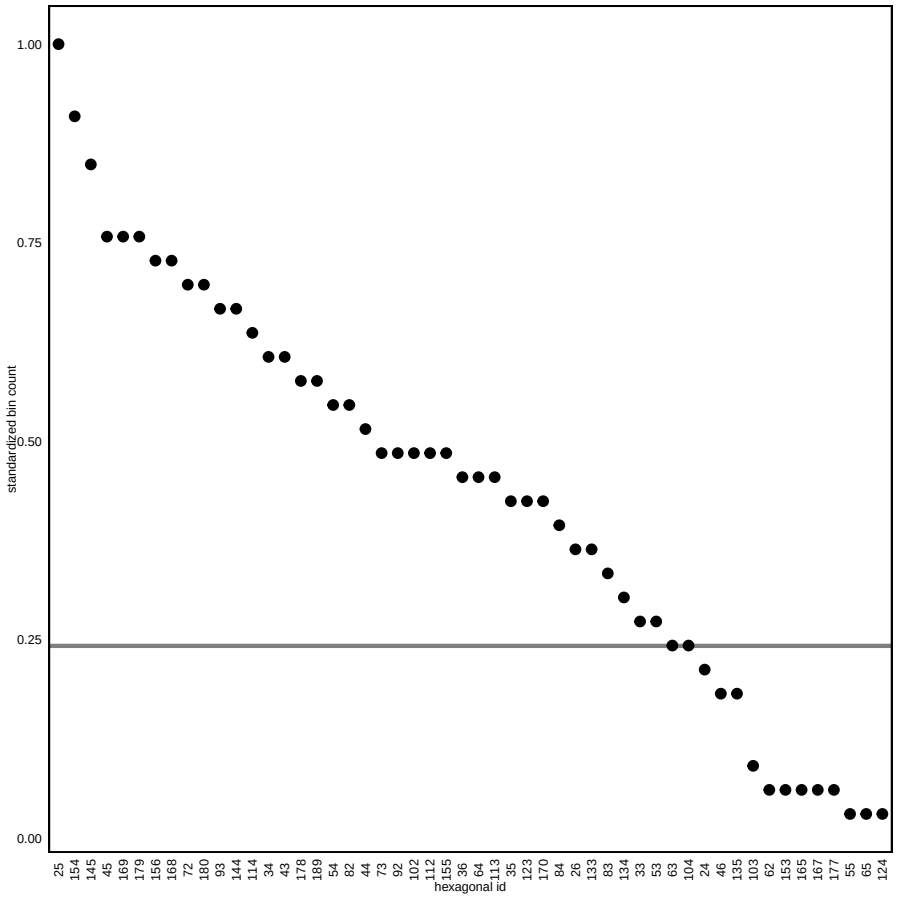


Figure 7: Distribution of standardize counts by hexagons.

The benchmark value for removing low-density hexagons ranges between 0 and 1. When analyzing how these benchmark values influence model performance, it's essential to observe the change in MSE as the benchmark value increases (see Figure 8). The MSE shows a gradual increase as the benchmark value progresses from 0 to 1. Evaluating this rate of increase is important. If the increment is not considerable, the decision might lean towards retaining low-density hexagons.

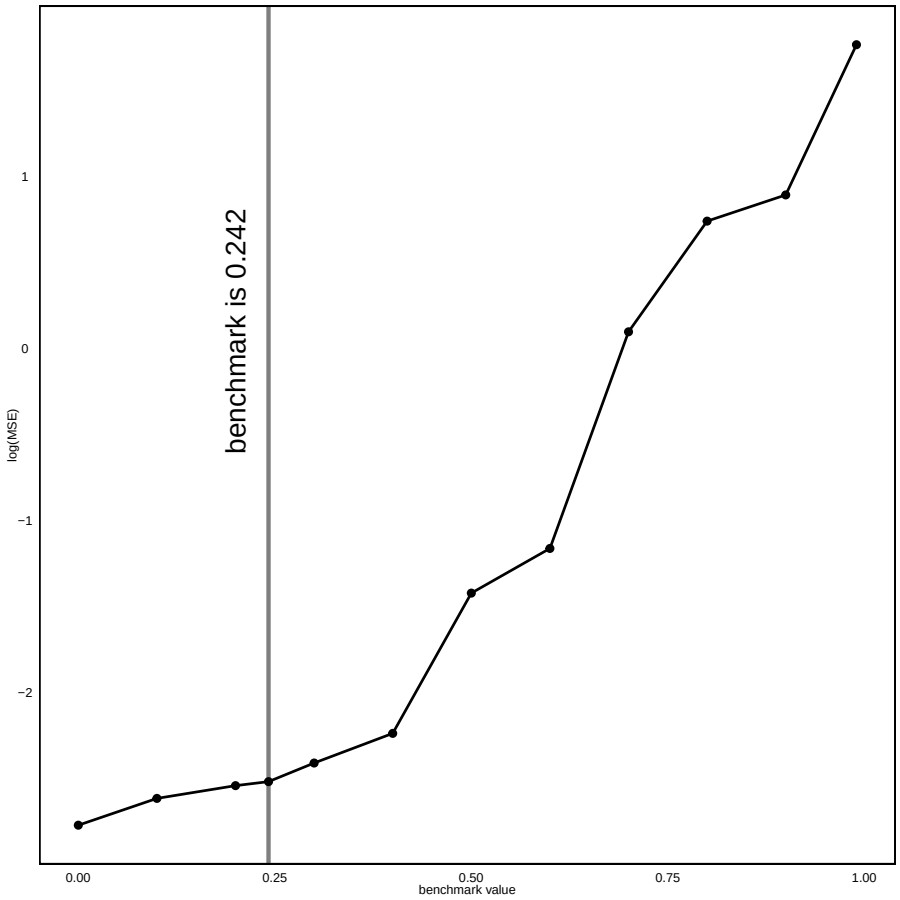


Figure 8: MSE from UMAP applied to S-curve dataset with diffferent bechmark choices. What is the effective bechmark value to remove low density hexagons? The residual plot have a steep slope at the end, indicating that a smaller bechmark value causes a small amount of MSE. Then, the slope gradually increases or level up, indicating that a higher bechmark value generates a higher MSE. Using the elbow method, it was observed that when the bechmark value is set to 0.242, the lowest MSE occurred.

Furthermore, selecting the benchmark value for removing low-density hexagons is important. Removing unnecessary bins may lead to the formation of long edges and an uneven 2-D model. Hence, rather than solely relying on the benchmark value to identify hexagons for removal, it's essential to consider the standard number of points in the neighboring hexagons of the identified low-density bins (see Figure 9 (b)). If neighboring bins also show low counts, only those bins will be removed. The remaining bins are used to construct the 2-D model.

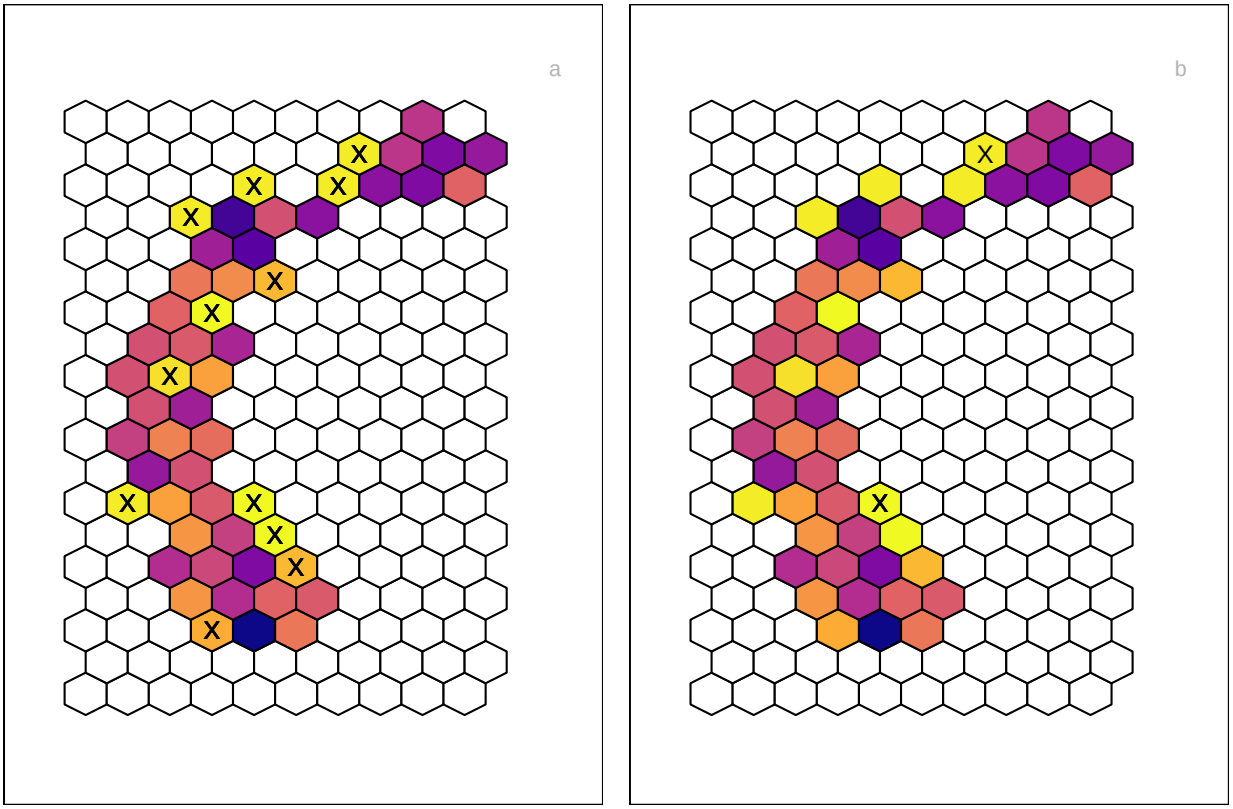


Figure 9: (a) Identification of low-density hexagons using a benchmark value of 0.242, and (c) Identification of low-density hexagons considering neighboring bins.

3.4.3 Removing long edges

To create a smooth 2-D representation (see Figure 2 (d)), it is necessary to remove edges that connect distant bin centroids in the triangular mesh. These edges only exist in the 2-D model and do not extend into p-D, so their removal does not impact the model in p-D. Although there are no specific criteria for determining the benchmark value to remove long edges, the following steps provide an approach to identifying a suitable threshold:

1. Plot the distribution of the 2D Euclidean distances (see Figure 10).
2. Identify the first largest difference between consecutive distance values.
3. Take the distance value corresponding to this difference as the benchmark value.

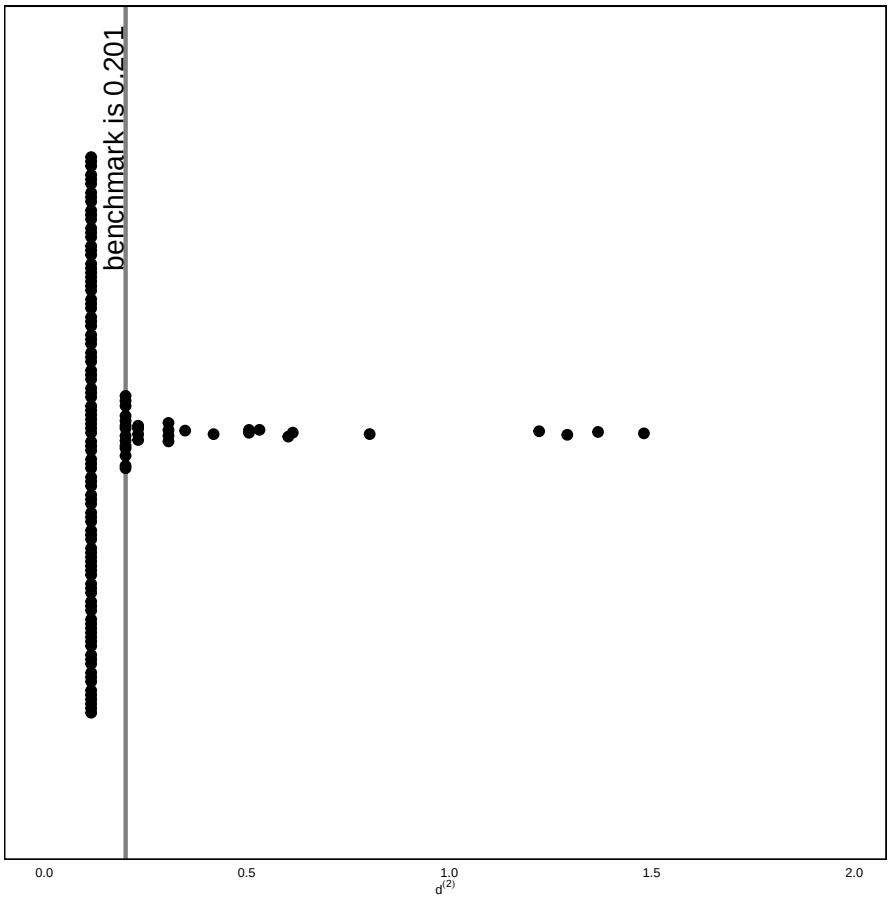


Figure 10: Distribution of 2-D Euclidean distances between bin centroids of the triangular mesh generated with S-curve UMAP data.

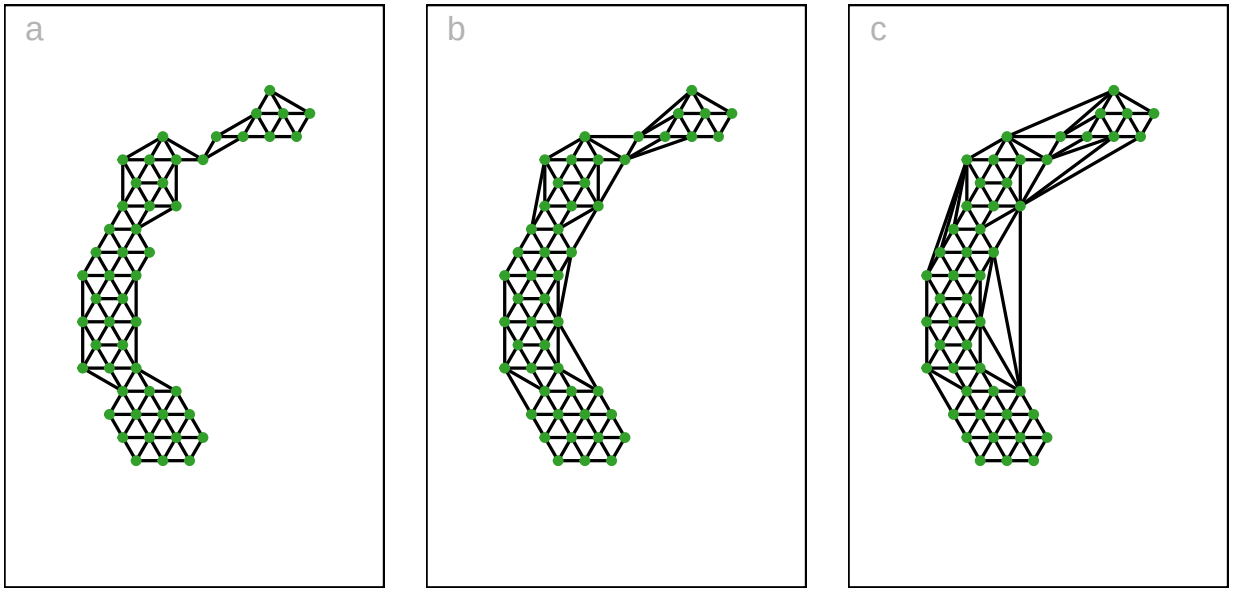


Figure 11: 2-D model generated for different benchamrk values to remove long edges: (a) benchmark = 0.201 (default), (b) benchmark = 0.35, and (c) benchmark = 1.1. What is the effective benchmark value to remove long edges?

3.5 Illustration on simulated data

In this section, the effectiveness of the algorithm is described using a simulated dataset. The dataset consists of five spherical Gaussian clusters in 4-D, with each cluster containing an equal number of points and the same within-cluster variation.

In the 2D layouts generated by various NLDR techniques, as shown in Figure 12, five well-separated clusters are shown. In tSNE (see Figure 12 (a)), these clusters appear closely. UMAP arranges all clusters in a parallel manner, with three aligned in one line and the other two in a separate line (see Figure 12 (b)). In contrast, PHATE shows two closely positioned clusters and three more distant ones (see Figure 12 (c)). In TriMAP, two clusters are close, though not as tightly as PHATE, while the other three are well-separated (see Figure 12 (d)). Finally, PaCMAP shows one central cluster and the remaining four spread out in different directions (see Figure 12 (e)).

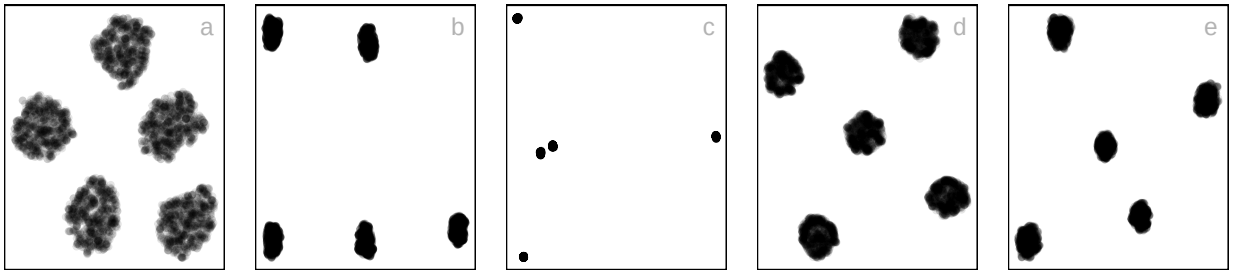


Figure 12: Five different NLDR representations of the same data. Different techniques and different parameter choices are used. Is there a best representation of the original data or are they all providing equivalent information?

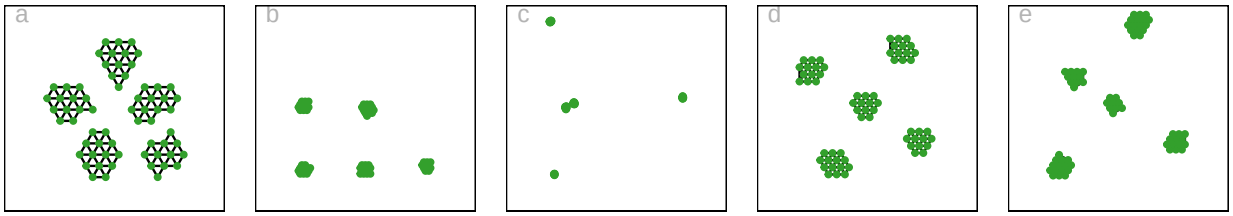


Figure 13: Model generated with five different NLDR methods in 2-D with approximately 65 non-empty bins in each.

To investigate which is the reasonable representation to visualize the five spherical Gaussian cluster data or all NLDR methods provide equivalent information, we visualize all the models in p-D space. Models from all NLDR methods show five well-separated clusters (see Figure 14, Figure 15, Figure 16, Figure 17, and Figure 18). This suggests that for the five Gaussian cluster dataset, all NLDR methods effectively preserve the global structure. tSNE displays clusters with varying densities, indicating their ability to capture within-cluster variation (see Figure 14). On the other hand, both UMAP, PHATE, PaCMAP and TriMAP show clusters with flat surfaces, suggesting a failure to capture within-cluster variation (see Figure 15, Figure 16, Figure 17 and Figure 18). Therefore, UMAP, PHATE, PaCMAP and TriMAP do not capture the local structure as effectively as other methods.

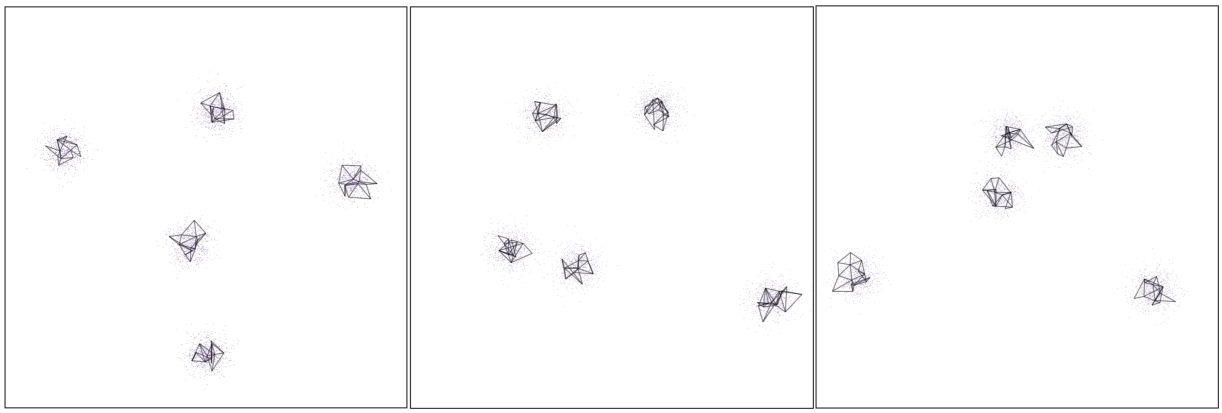


Figure 14: Screen shots of the **langevitour** of the five Gaussian clusters dataset, shows the model with tSNE in p-D, a video of the tour animation is available at (https://youtu.be/RASEE7N5MbM).

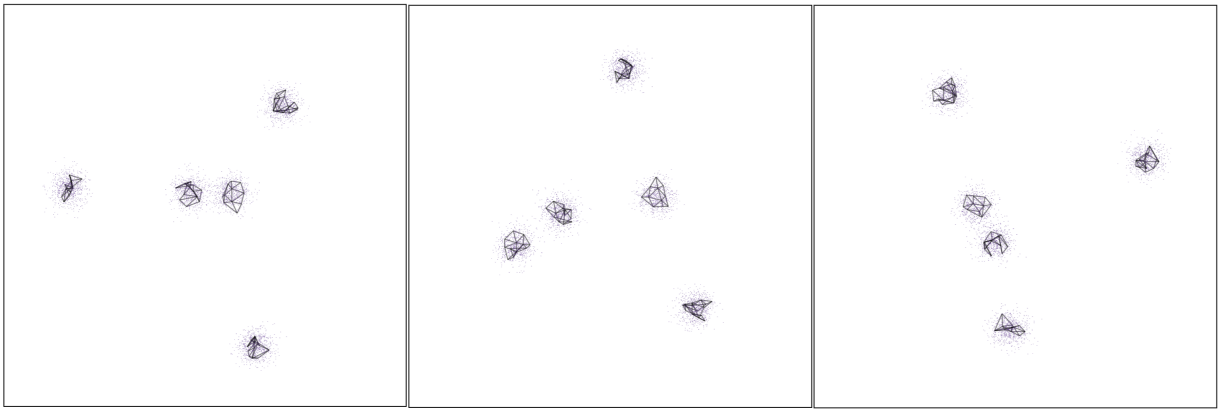


Figure 15: Screen shots of the **langevitour** of the five Gaussian clusters dataset, shows the model with UMAP in p-D, a video of the tour animation is available at (https://youtu.be/iG4bCPkJilw).

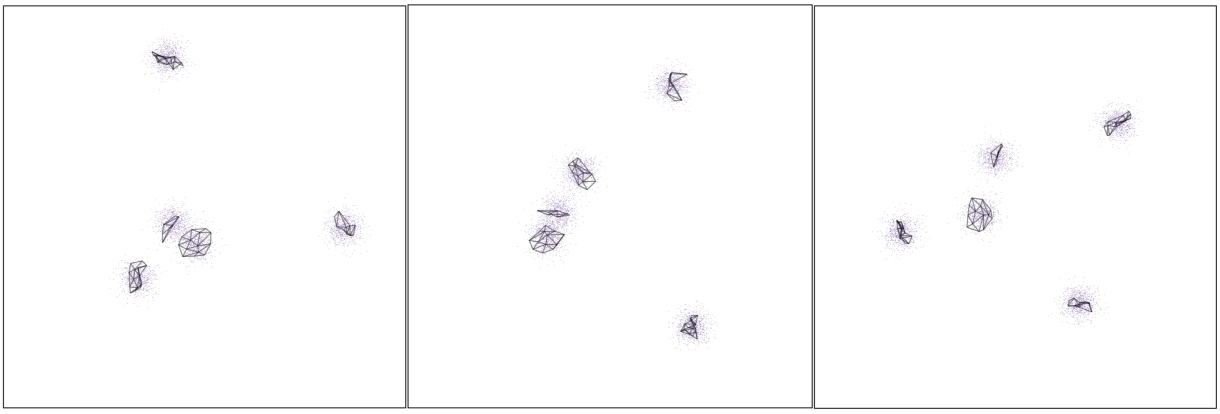


Figure 16: Screen shots of the **langevitour** of the five Gaussian clusters dataset, shows the model with PHATE in p-D, a video of the tour animation is available at (https://youtu.be/L_PVLGwfOS0).

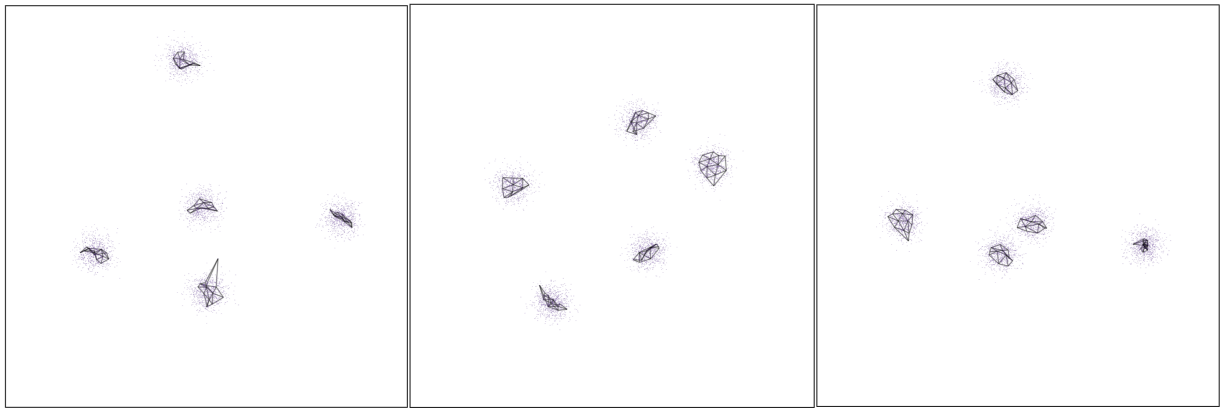


Figure 17: Screen shots of the **langevitour** of the five Gaussian clusters dataset, shows the model with PaCMAP in p-D, a video of the tour animation is available at (https://youtu.be/z07cKXi8EJQ).

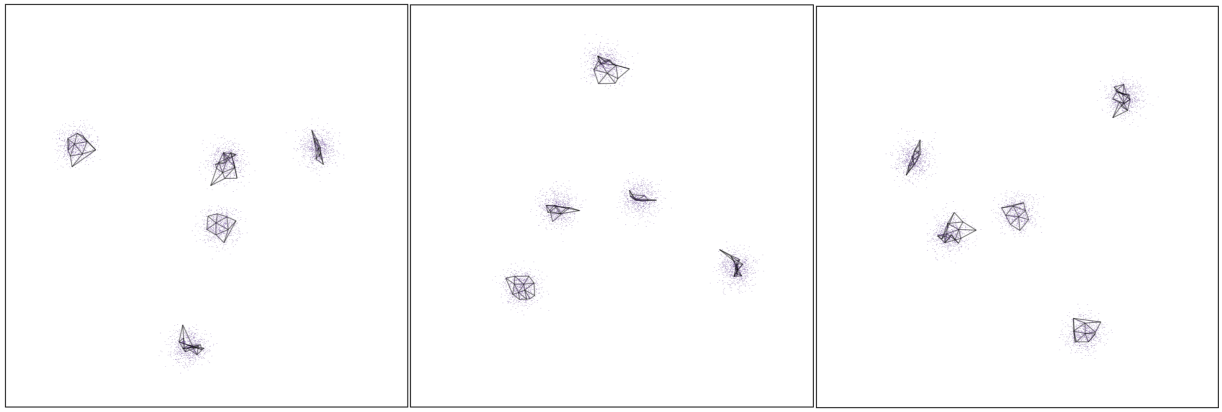


Figure 18: Screen shots of the **langevitour** of the five Gaussian clusters dataset, shows the model with TriMAP in p-D, a video of the tour animation is available at (https://youtu.be/Chs1lYAoX2w).

When compare the NLDR representations and generated models, tSNE with perplexity 61 appears to be a reasonable representation for visualizing the five Gaussian cluster dataset. This is supported by investigating the model generated with tSNE in the data space, which provides evidence that it preserves both local and global structures. Also, the NLDR representation with tSNE shows five well-separated clusters.

4 Applications

4.1 pbmc

In the field of single-cell studies, a common analytical task involves clustering to identify groups of cells with similar expression profiles. Analysts often turn to NLDR techniques to verify and identify these clusters and explore developmental trajectories. To illustrate the importance of NLDR techniques and parameter selection in identifying clusters, Human

Peripheral Blood Mononclear Cells (PBMC3k) dataset (Chen et al. 2023) is used. In a study by Chen et al. (2023), this dataset was used to demonstrate how UMAP represents clusters (see Figure 19). As shown in Figure 19, there are three distant and well-separated clusters.

Figure 19: 2-D layout from UMAP applied for the PBMC3k dataset. Is this a best representation of the original data? The parameter setting is n_neighbors = 30, min_dist = 0.3.

To determine whether the UMAP representation with the parameter choice suggested by Chen et al. (2023) preserves the original data structure, we visualize the model constructed with UMAP overlaid on the p-D data. The figures in Figure 21 show three well-separated clusters, indicating that the suggested UMAP representation preserves the global structure (see Figure 19). However, as shown in Figure 21, these clusters are close to each other in p-D. Also, non-linear continuity patterns and high-density patches within the clusters are observed (see Figure 21). Therefore, the suggested UMAP representation (see Figure 19) does not accurately preserve the local structure of the PBMC3k dataset.

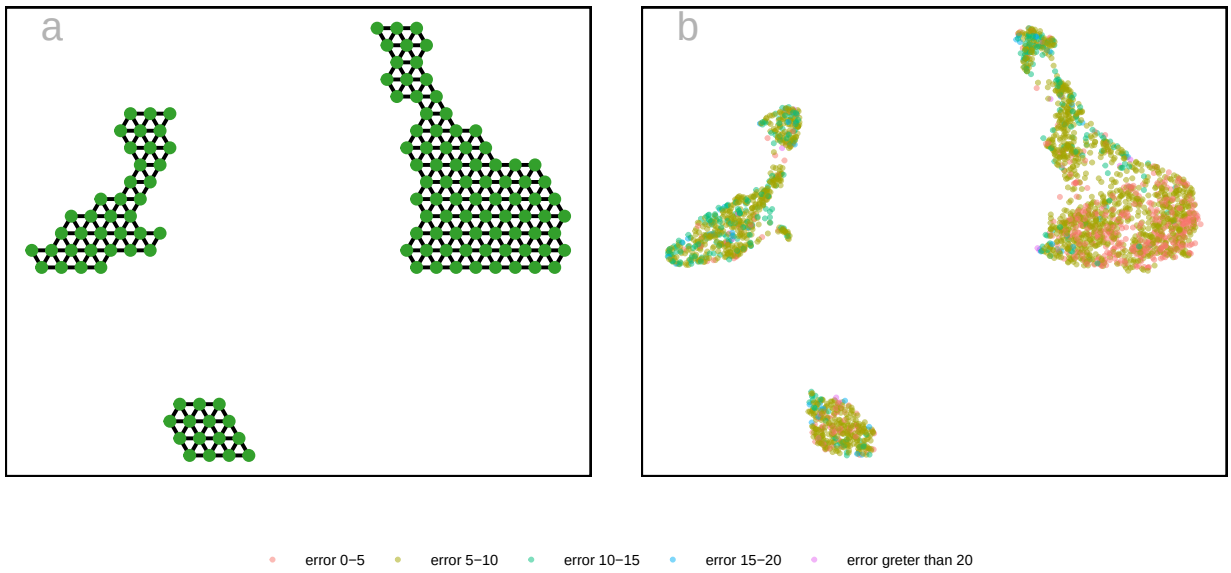


Figure 20: (a) Model generated in 2-D with UMAP, and (b) p-D model error in 2-D. The 2-D model shows three well-separated distant clusters. The p-D model errors are distributed along clusters.

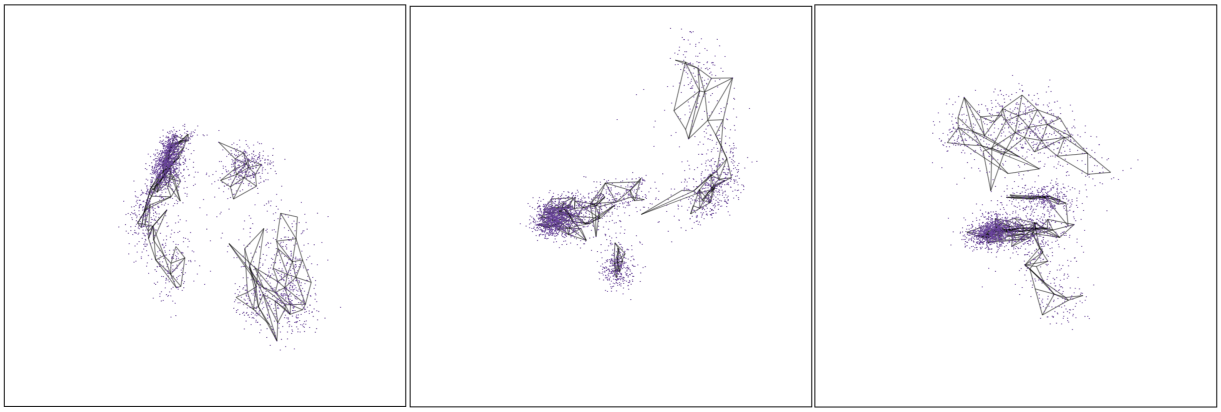


Figure 21: Screen shots of the **langevitour** of the PBMC3k data set, shows the model-in-data space, a video of the tour animation is available at (https://youtu.be/VqqWuE0Jj6A).

In order to find a reasonable NLDR representation for the PBMC3k dataset, we calculated the absolute error for different numbers of non-empty bins using various NLDR techniques and different parameter settings (see Figure 22). After analyzing the results, we found that tSNE with a perplexity set to 30 had the lowest error when the number of non-empty bins was 137. Therefore, tSNE with a perplexity of 30, which is the default parameter setting, is considered as a reasonable representation for the PBMC3k dataset.

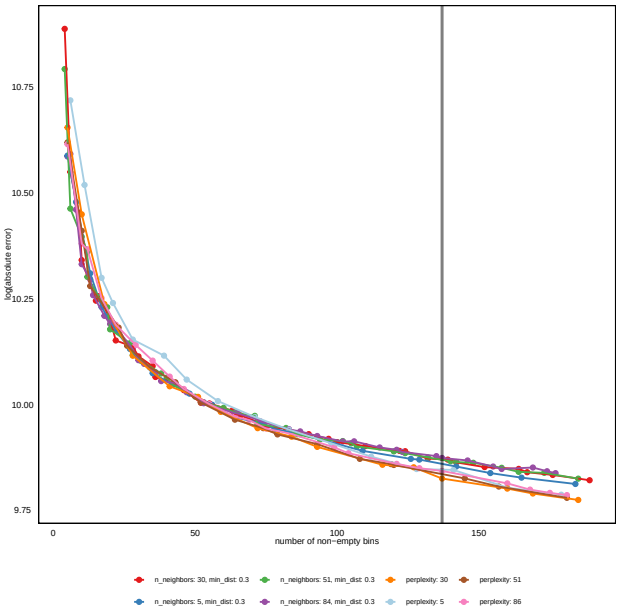


Figure 22: Absolute error from UMAP and tSNE applied to training PBMC3k dataset with diffferent parameter choices. What is the best parameter choice to create the model? The residual plot have a steep slope at the beginning, indicating that a smaller number of non-empty bins causes a larger amount of error. Then, the slope gradually declines or level off, indicating that a higher number of non-empty bins generates a smaller error. Using the elbow method, it was observed that when the number of non-empty bins is set to 137, the lowest error occurred with the parameters perplexity: 30.

As shown in Figure 23, there are three well-separated clusters, although they are located close to each other. Additionally, non-linear structures can also be observed within the clusters (see Figure 24 (a)). In this manner, tSNE was able to capture the data structure for the PBMC3k dataset that UMAP failed to do.

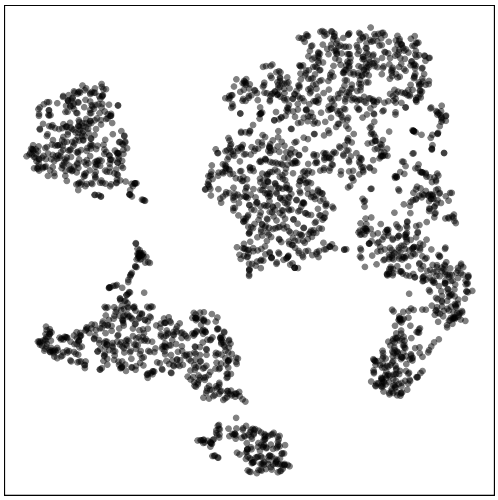


Figure 23: 2-D layout from tSNE applied for the PBMC3k dataset. Is this a best representation of the original data? The parameter setting is perplexity=30.

We then fit the model for tSNE, and visualize the resultant model in the p-D data space. The model shows a quirk, as shown in Figure 25. All three clusters are connected by an edge except the small and large clusters. Because the clusters are so close in 2-D, they attempt to maintain the structure in p-D as well. This is evident that tSNE with perplexity 30 provides a reasonable representation of PBMC3k data.

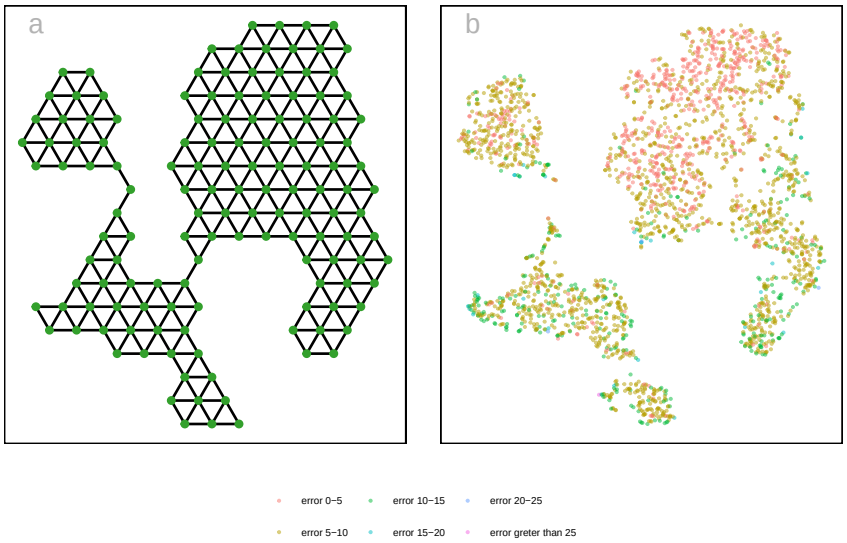


Figure 24: (a) Model generated in 2-D with tSNE, and (b) p-D model error in 2-D. The 2-D model shows three well-separated distant clusters. The p-D model errors are distributed along clusters, but most low p-D model errors present in the large cluster.

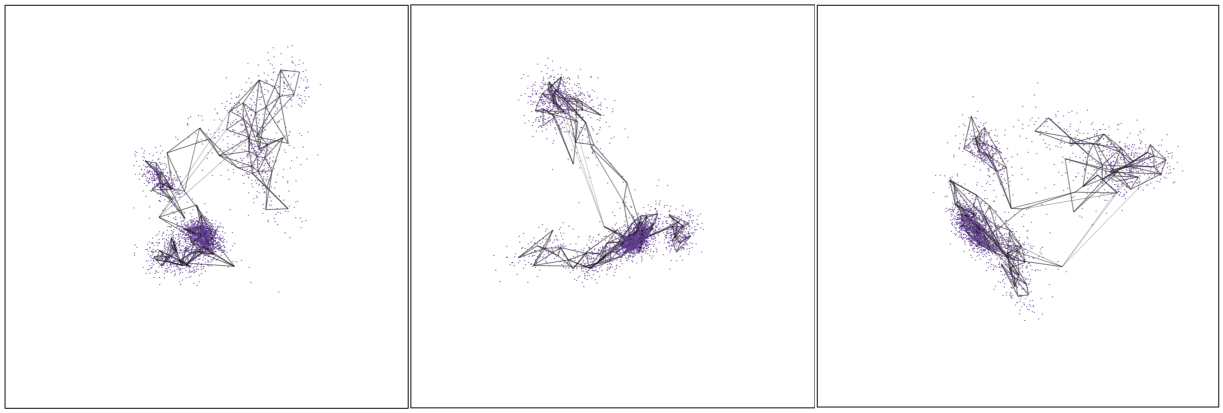


Figure 25: Screen shots of the **langevitour** of the PBMC3k data set, shows the model in high-D, a video of the tour animation is available at (https://youtu.be/5Y1hE4i7N2k).

4.2 digits: 1

The MNIST dataset consists of grayscale images of handwritten digits (LeCun & Cortes 2010). Wang et al. (2021) used this dataset to demonstrate how PaCMAP preserves local structure. We selected the 2-D embedding of PaCMAP for the handwritten digit 1 to assess whether this is a reasonable representation using our method. As shown in Figure 27, the angle of the digit 1 images varies along the 2-D structure.

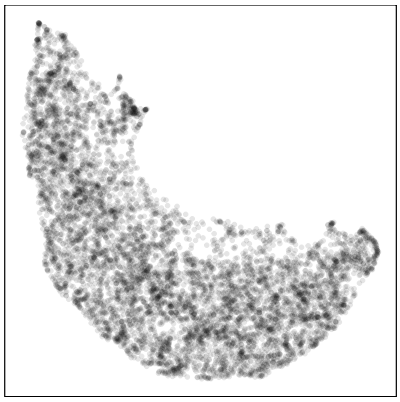


Figure 26: 2-D layout from PaCMAP applied for the digit 1 of the MNIST dataset. Is this the best representation of the digit 1? The parameter setting is n_components=2, n_neighbors=10, init=random, MN_ratio=0.9, FP_ratio=2.0.

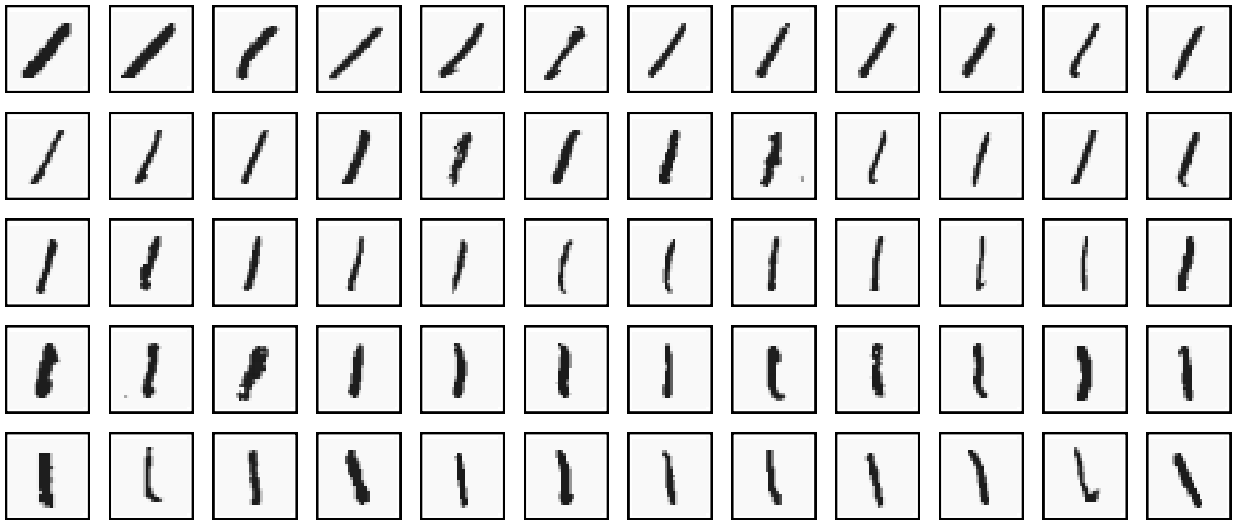


Figure 27: Images of the handwritten digit 1 are ordered from the bottom-right to the top-left of the 2-D structure. The angle of the digit varies along this structure. Images at the bottom-right of the 2-D layout show the digit 1 angled more to the right, while images at the top-left show the digit 1 angled more to the left. This demonstrates how the angle changes from right to left along the 2-D structure.

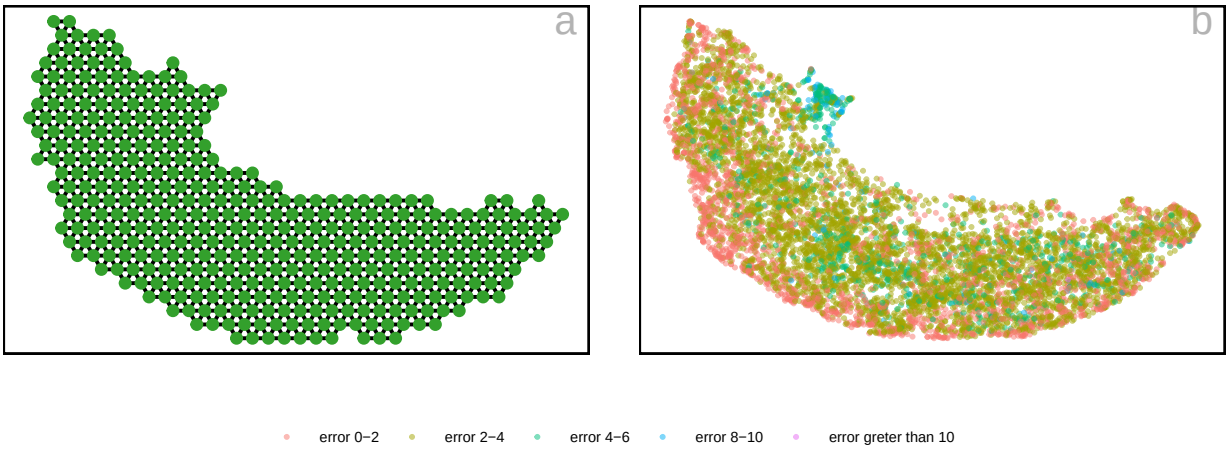
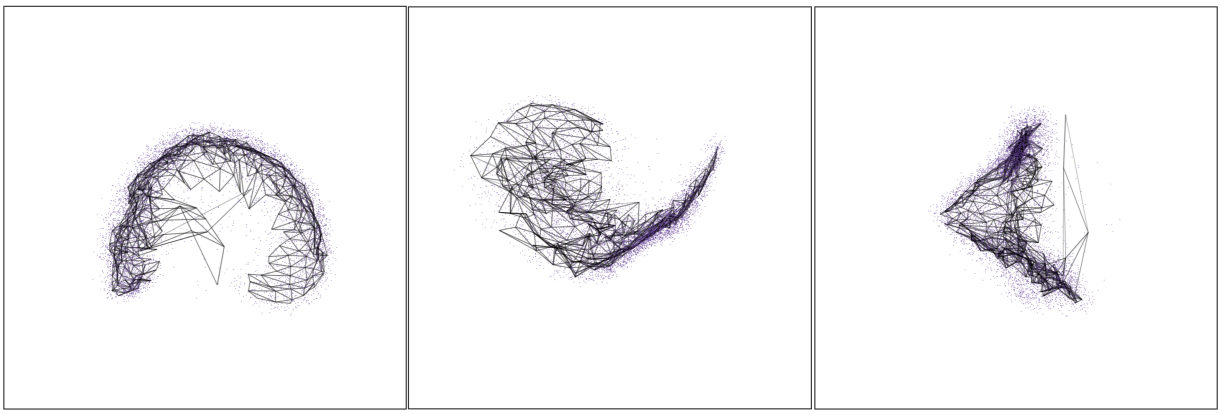


Figure 28: (a) Model generated in 2-D, and (b) p-D model error in 2-D. The 2-D model shows a non-linear continuous structure. Most low p-D model errors are distributed along the lower edge of the 2-D structure, while most high p-D model errors are concentrated along the upper edge.

According to Figure 29a, the non-linear continuous structure observed in the 2-D representation of PaCMAP (see Figure 26) is also visible when visualizing the model overlaid on the data space. This indicates that PaCMAP accurately captures the structure of the p-D data. Additionally, the model shows a twisted pattern within the non-linear structure in p-D space (see Figure 29b), which is an additional pattern not visible in the 2-D representation (see Figure 26). Furthermore, as shown in Figure 29c, some long edges exist in the p-D space that are not recognized as long edges in the 2-D representation. However, PaCMAP is a reasonable 2-D representation of MNIST digit 1 data. Because PaCMAP preserves the local structure.



(a) (b) (c)

Figure 29: Screen shots of the **langevitour** of the MNIST digit 1 data set, shows the model-in-data space, a video of the tour animation is available at (https://youtu.be/zcg_GXBmqjA).

There are certain data points that exhibit high error rates due to their deviation from the usual p-D data structure, which makes them anomalies (see Figure 28 (b)). These

anomalies can be classified into two types: those that are anomalies within the non-linear structure and those that lie outside of it. The images associated with high model error points within the non-linear structure display different patterns of the digit 1, as shown in Figure 30 (a). However, when comparing these images to the ones found outside of the non-linear structure, it becomes evident that the latter display different patterns of the digit 1 (see Figure 30 (b)).

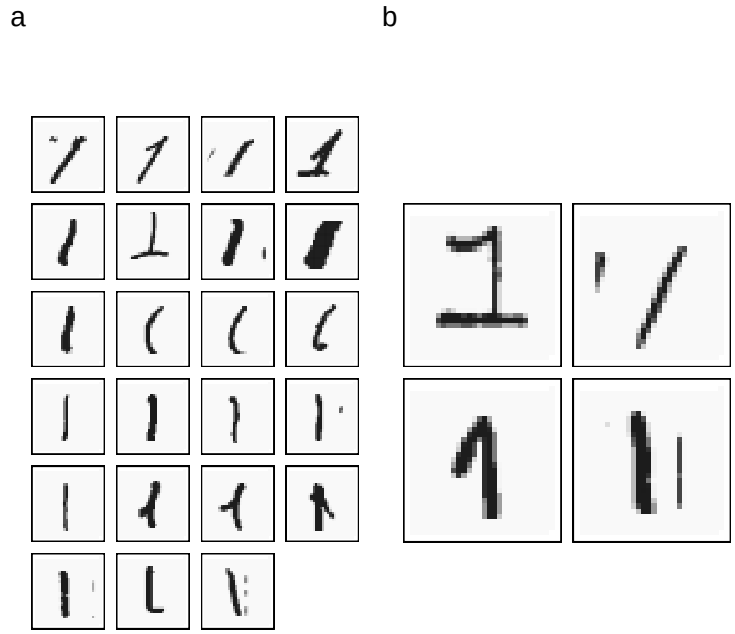


Figure 30: Some images of handwritten digit 1 which occur high model error (a) within the non-linear strcuture, and (b) outside the non-linear structure. The images shows different patterns of digit 1.

5 Discussion

- Summarise contributions
- Explain where it is expected or not expected to work, eg higher dimensional relationships
- Human behaviour, the desire to have more certainty, and a tendency to prefer the well-separated views
- Predicting new observations in k-D
- Extending layouts beyond k-D, when 2D is clearly inadequate.
- Diagnostic app to explore differences in distances
- What might be useful enhancements

References

Amid, E. & Warmuth, M. K. (2022), 'Trimap: Large-scale dimensionality reduction using triplets'.

Borg, I. & Groenen, P. J. F. (2005), *Modern Multidimensional Scaling Theory and Applications*, Springer, New York.

Chen, Z., Wang, C., Huang, S., Shi, Y. & Xi, R. (2023), 'Directly selecting differentially expressed genes for single-cell clustering analyses', *bioRxiv* .
URL: *https://www.biorxiv.org/content/early/2023/07/29/2023.07.26.550670*

Coifman, R., Lafon, S., Lee, A., Maggioni, M., Nadler, B., Warner, F. & Zucker, S. (2005), 'Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps', *Proceedings of the National Academy of Sciences of the United States of America* **102**, 7426–31.

Harrison, P. (2023), 'langevitour: Smooth interactive touring of high dimensions, demonstrated with scrna-seq data', *The R Journal* **15**, 206–219. https://doi.org/10.32614/RJ-2023-046.

Hart, C. & Wang, E. (2022), *detourr: Portable and Performant Tour Animations*. R package version 0.1.0.
URL: *https://casperhart.github.io/detourr/*

Johnstone, I. M. & Titterington, D. M. (2009), 'Statistical challenges of high-dimensional data', *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* **367**(1906), 4237–4253.
URL: *https://royalsocietypublishing.org/doi/abs/10.1098/rsta.2009.0159*

Jolliffe, I. (2011), *Principal Component Analysis*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 1094–1096.
URL: *https://doi.org/10.1007/978-3-642-04898-2_455*

Jöreskog, K. G. (1969), 'A general approach to confirmatory maximum likelihood factor analysis', *Psychometrika* pp. 183–202.
URL: *https://doi.org/10.1007/BF02289343*

Laa, U., Cook, D. & Lee, S. (2022), 'Burning sage: Reversing the curse of dimensionality in the visualization of high-dimensional data', *J. Comput. Graph. Stat.* **31**(1), 40–49.
URL: *https://doi.org/10.1080/10618600.2021.1963264*

LeCun, Y. & Cortes, C. (2010), 'MNIST handwritten digit database'.
URL: *http://yann.lecun.com/exdb/mnist/*

Lee, S., Cook, D., da Silva, N., Laa, U., Wang, E., Spyrison, N. & Zhang, H. S. (2021), 'A review of the state-of-the-art on tours for dynamic visualization of high-dimensional data'.

McInnes, L. & Healy, J. (2018), 'Umap: Uniform manifold approximation and projection for dimension reduction', *ArXiv* **abs/1802.03426**.

Moon, K. R., van Dijk, D., Wang, Z., Gigante, S. A., Burkhardt, D. B., Chen, W. S., Yim, K., van den Elzen, A., Hirn, M. J., Coifman, R. R., Ivanova, N. B., Wolf, G. & Krishnaswamy, S. (2019), 'Visualizing structure and transitions in high-dimensional biological data', *Nature Biotechnology* **37**, 1482 – 1492.

Saeed, N., Nam, H., Haq, M. I. U. & Muhammad Saqib, D. B. (2018), 'A survey on multidimensional scaling', *ACM Comput. Surv.* **51**(3).
URL: *https://doi.org/10.1145/3178155*

Silva, V. & Tenenbaum, J. (2002), 'Global versus local methods in nonlinear dimensionality reduction', *Advances in neural information processing systems* **15**.

van der Maaten, L. & Hinton, G. E. (2008), 'Visualizing data using t-sne', *Journal of Machine Learning Research* **9**, 2579–2605.

Wang, Y., Huang, H., Rudin, C. & Shaposhnik, Y. (2021), 'Understanding how dimension reduction tools work: An empirical approach to deciphering t-sne, umap, trimap, and pacmap for data visualization', *Journal of Machine Learning Research* **22**(201), 1–73.
URL: *http://jmlr.org/papers/v22/20-1061.html*

Wickham, H., Cook, D. & Hofmann, H. (2015), 'Visualizing statistical models: Removing the blindfold', *Statistical Analysis and Data Mining: The ASA Data Science Journal* **8**(4), 203–225.
URL: *https://onlinelibrary.wiley.com/doi/abs/10.1002/sam.11271*

Wickham, H., Cook, D., Hofmann, H. & Buja, A. (2011), 'tourr: An r package for exploring multivariate data with projections', *Journal of Statistical Software* **40**(2), 1—18.
URL: *http://www.jstatsoft.org/v40/i02/*