

# Dimension Reduction with Locally Adjusted Graphs

**Yingfan Wang\***, **Yiyang Sun\***, **Haiyang Huang\***, **Cynthia Rudin**

Duke University

{yingfan.wang, yiyang.sun, haiyang.huang, cynthia.rudin}@duke.edu

## Abstract

Dimension reduction (DR) algorithms have proven to be extremely useful for gaining insight into large-scale high-dimensional datasets, particularly finding clusters in transcriptomic data. The initial phase of these DR methods often involves converting the original high-dimensional data into a graph. In this graph, each edge represents the similarity or dissimilarity between pairs of data points. However, this graph is frequently suboptimal due to unreliable high-dimensional distances and the limited information extracted from the high-dimensional data. This problem is exacerbated as the dataset size increases. If we reduce the size of the dataset by selecting points for a specific sections of the embeddings, the clusters observed through DR are more separable since the extracted subgraphs are more reliable. In this paper, we introduce LocalMAP, a new dimensionality reduction algorithm that dynamically and locally adjusts the graph to address this challenge. By dynamically extracting subgraphs and updating the graph on-the-fly, LocalMAP is capable of identifying and separating real clusters within the data that other DR methods may overlook or combine. We demonstrate the benefits of LocalMAP through a case study on biological datasets, highlighting its utility in helping users more accurately identify clusters for real-world problems.

**Code** — <https://github.com/williamsyy/LocalMAP>

## 1 Introduction

Dimension reduction (DR) is an important data visualization strategy for understanding the structure of complicated high-dimensional datasets. DR is used extensively in image, text, and biomedical datasets, particularly -omics (Cao et al. 2019; Becht et al. 2019; Amezquita et al. 2020; Dries et al. 2021; Atitey, Motsinger-Reif, and Anchang 2024; Böhm, Berens, and Kobak 2023; Mu, Bhat, and Viswanath 2017; Raunak, Gupta, and Metze 2019). Beginning in the original high-dimensional space, DR methods typically first abstract the data into a *graph*, with each edge representing either similarity (positive edge) or dissimilarity (negative edge). This graph is then used to optimize the low-dimensional embedding by applying attractive forces along the positive edges and repulsive forces along the negative edges. Clearly, the quality of

the graph, which connects the original high-dimensional data to the low-dimensional embedding, is critical to this process. However, given the complex nature of high-dimensional data, it is challenging to construct a graph that accurately represents all patterns and structures within the data, and graphs are usually sub-optimal.

In this work, we provide two key insights as to why the graphs could be flawed. First, we found that high-dimensional distances become less informative due to the curse of dimensionality, where measurements of similarity and dissimilarity determined using high-dimensional distances do not necessarily reflect similarities and dissimilarities along the data manifold. This issue is more pronounced in DR methods that select a small group of nearest neighbors (NNs) to form positive edges of the graph and apply strong attractive forces along these edges. When a “false” positive edge is constructed based on an inaccurate similarity measure between a pair of points that are actually dissimilar, strong attractive forces will pull them closer. If there are many such false positive edges, the DR method will generate overlapping clusters without clear boundaries, even if the data have distinct clusters. Since DR methods are unsupervised, the user would not be able to determine that distinct clusters exist – they would be blurred together in the DR result. As we show in this work, eliminating these false positive edges can dramatically improve DR projections.

Our second insight is that missing edges (lack of negative “further pair” edges between far points in high-dimensional space) also contribute to unwanted overlapping of clusters, since such edges help to define boundaries between clusters. However, as the scale of the dataset increases, the set of negative edges become both insufficient and less effective, as discussed in Section 4.2. Some of the most important applications of DR (single cell, transcriptomics) generate large high-dimensional datasets with many clusters, so it is important that there are enough further pair (FP) edges to separate them.

Based on the two insights discussed above, we propose LocalMAP, a new DR algorithm that dynamically and locally adjusts the graph *during dimension reduction* to address the aforementioned issues with graph construction. LocalMAP *detects false positive edges and removes them*, using ideas similar to outlier detection in robust statistics. This allows clusters to separate. It also *adds more further pair edges*

\*These authors contributed equally.

Copyright © 2025, Association for the Advancement of Artificial Intelligence ([www.aaai.org](http://www.aaai.org)). All rights reserved.

dynamically, allowing crisper cluster boundaries to appear. Together, these ideas within LocalMAP produce clear, crisp separated clusters where other methods fail.

Figure 1 shows the results of several high-quality DR approaches, including t-SNE (van der Maaten and Hinton 2008), UMAP (McInnes, Healy, and Melville 2018) and PaCMAp (Wang et al. 2021), on the MNIST dataset where there are 10 separated clusters in the high-dimensional space. Algorithms t-SNE, UMAP and PaCMAp generate DR embeddings without clear boundaries between clusters, while LocalMAP generates a high-quality DR embedding with well-defined boundaries that are visible even without class information provided to the algorithm.

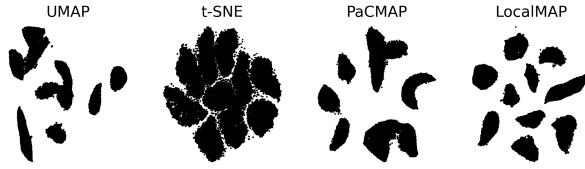


Figure 1: DR embeddings on MNIST dataset which contains 10 digit classes. Our LocalMAP method is on the right. The colored embeddings with true labels are shown in Figure 5.

In this work, we provide more evidence for the insights discussed above, in terms of both theory based on simple clustering models and empirical evidence. We also provide LocalMAP and a comprehensive evaluation of it, based on Huang et al. (2022). We provide case studies showing that LocalMAP is able to find true clusters – and not false clusters – more reliably than other DR approaches.

## 2 Related Work

While DR methods that preserve *global structure* date back to 1901 with PCA (Pearson 1901), multidimensional scaling (Torgerson 1952), and non-negative matrix factorization (Lee and Seung 1999), methods that preserve *local structure* have become indispensable and ubiquitous in numerous applications, particularly in -omics research. Local methods include Isomap (Tenenbaum, de Silva, and Langford 2000), Local Linear Embedding (LLE) (Roweis and Saul 2000), Laplacian Eigenmap (Belkin and Niyogi 2001), and more recent Neighborhood Embedding (NE) algorithms like t-SNE (van der Maaten and Hinton 2008), LargeVis (Tang et al. 2016), UMAP (McInnes, Healy, and Melville 2018), PaCMAp (Wang et al. 2021), TriMAP (Amid and Warmuth 2019), NCVIS (Artemenkov and Panov 2020), h-NNE (Sarfraz et al. 2022), Neg-t-SNE (Damrich et al. 2023) and InfoNC-t-SNE (Damrich et al. 2023) and many others (Van Assel et al. 2022; Zu and Tao 2022). Global methods typically are not able to preserve separations between clusters or clearly display the data manifold, whereas local methods can sometimes do so; thus, they provide a unique perspective on the data that is difficult to gain in any other way.

Because DR methods are unsupervised, there is no common objective function like there would be for supervised learning (e.g., classification error). However, there are prin-

ciples that reliable DR loss functions typically obey (Wang et al. 2021). We will discuss those in Appendix C, as LocalMAP obeys these principles based on what it inherits from PaCMAp.

There are works that discuss how DR methods’ behavior is affected by different components of the DR algorithm. For example, there are several papers discussing the challenges of tuning parameters and applying these methods in practice (Wattenberg, Viégas, and Johnson 2016; Cao and Wang 2017; Nguyen and Holmes 2019; Belkina et al. 2019; Kobak and Lindereman 2021). Wang et al. (2021); Böhm, Berens, and Kobak (2020) also discuss how changing different graph components and loss functions affect DR methods’ behavior.

*None of the above approaches adjusts the graph itself during DR optimization.* The graph is typically considered to be fixed as ground truth; the graph information is analogous to the labels for classification tasks that are also considered ground truth. However, recent work has found improvements in classification performance by identifying possibly mislabeled points and omitting them (Chen et al. 2019; Han et al. 2018; Northcutt, Athalye, and Mueller 2024), showing that there is value in identifying and removing labels that appear to be wrong *during the classification process*. There is also a field of robust statistics that identifies and omits outliers that would wreck performance, and aims to ensure results are robust to assumptions on the data distribution (Martin et al. 2018; Li, Socher, and Hoi 2020). Our work is thus unique in extending these types of idea to DR, and not trusting every graph element as if it were correct. As discussed, we know the graph is probably wrong when the Euclidean distance is not the same as the distance along the data manifold (geodesic distance).

## 3 Review of PaCMAp

The proposed LocalMAP algorithm starts from a (faulty) embedding that is already formed. We first review PaCMAp since LocalMAP can start after its first two phases. Consider a dataset  $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$  consisting of  $n$  points in a high-dimensional space. PaCMAp seeks to find a low-dimensional embedding  $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n\}$ , where each  $\mathbf{y}_i$  corresponds to  $\mathbf{x}_i$ . PaCMAp first constructs a graph in high-dimensional space with three kinds of pairs: NN pairs (near neighbors in the high-dimensional space), MN pairs (mid-near pairs, close but not as close as neighbors), and FP pairs (further point pairs, far in the high-dimensional space). Optimization of the low-dimensional embedding  $\mathbf{Y}$  is performed using a simple objective:

$$\begin{aligned} Loss^{\text{PaCMAp}} = & w_{\text{NN}} \cdot \sum_{(i,j): \text{NN}} \frac{\tilde{d}_{ij}}{C_{\text{Med}} + \tilde{d}_{ij}} + \\ & w_{\text{MN}} \cdot \sum_{(i,k): \text{MN}} \frac{\tilde{d}_{ik}}{C_{Lg} + \tilde{d}_{ik}} + w_{\text{FP}} \cdot \sum_{(i,l): \text{FP}} \frac{1}{1 + \tilde{d}_{il}} \end{aligned} \quad (1)$$

where  $\tilde{d}_{ij} = d_{ij}^2 + 1 = \|\mathbf{y}_i - \mathbf{y}_j\|^2 + 1$ .  $C_{Lg}$  and  $C_{\text{Med}}$  are nontunable parameters controlling the scale of the embedding, set at  $\sim 10K$  and  $\sim 10$ . The weights  $w_{\text{NN}}$ ,  $w_{\text{MN}}$ , and  $w_{\text{FP}}$  are adjusted to balance attraction and repulsion. Neither the

weights nor the parameters should be modified by users; they perform well across datasets (Wang et al. 2021).

## 4 Dynamically and Locally Adjusted Graph

### 4.1 Insight 1: False positive nearest neighbor edges pull nearby clusters together, losing clear boundaries between them

Let us consider an experiment to illustrate this. DR methods apply attractive forces along the NN edges (between points that are close in the high-dimensional space) and apply repulsive forces along FP edges (points that are far in the high-dimensional space). An NN edge between two points is *preserved* if these points are placed nearby in the low-dimensional embedding. Given the limited capacity of low-dimensional spaces and the complex nearest-neighbor relationships in high-dimensional datasets, it is clear that not all NN edges can be preserved during dimensionality reduction. Further, we do not want to preserve all high-dimensional NNs, since the Euclidean distance neighbors are not always the true neighbors along the actual data manifold. Therefore, the question becomes how to identify which NNs to preserve.

The preservation of specific NN edges depends on intricate dynamics during the embedding optimization, which utilizes the graph extracted from the high-dimensional data. For a pair of NNs  $(i, j)$ , if they share a greater number of common neighbors, which provide attraction along NN paths, it is more likely that these two points will be positioned close to each other in the low-dimensional space. We hypothesize that these dynamics sometimes correct erroneous edges generated by unreliable high-dimensional distances. If true, this hypothesis implies we can adjust the graph dynamically based on this information and further optimize the embedding using the revised graph.

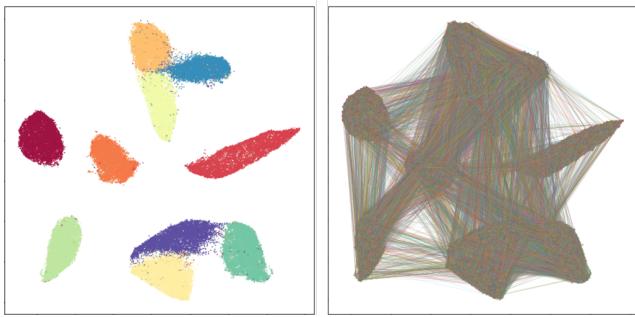


Figure 2: Visualization of NN edge connections of PaCMAP embedding on MNIST dataset.

Figure 2 presents a PaCMAP embedding of the MNIST dataset on the left and the corresponding NN edges' visualization on the right. The figure highlights a substantial number of NN edges bridging distinct clusters that are widely separated. When two points connected by such an NN edge are distant in the final embedding, it suggests that the dynamics of the optimization process have separated them, indicating

that the NN edge might be erroneous. Despite this, the erroneous NN edge continues to exert an attractive force, pulling these two points towards each other. If numerous such NN edges exist between two clusters, they can lead to clusters being falsely connected, rather than clearly delineated with distinct boundaries.

### 4.2 Insight 2: Insufficient and ineffective negative further point edges fail to create clear boundaries between nearby clusters, particularly in large datasets

Ideally, the repulsive forces along the FP edges should separate nearby clusters. However, when the dataset becomes larger, with more clusters and larger samples, it is more likely that the sampled FP edges in DR algorithms are insufficient.

**Theorem 1.** *Assume all points between two clusters are approximately equidistant, so that the probability of constructing a positive pair between points from these clusters is constant. The ratio between the number of NN edges to the number of FP edges of PaCMAP between two clusters increases with the number of data points in a dataset.*

*Proof.* Consider a dataset with  $n$  data points distributed across  $m$  clusters  $C_1, C_2, \dots, C_m$ , where each cluster  $C_i$  contains  $n_i$  data points ( $n_1 + n_2 + \dots + n_m = n$ ). For any two clusters  $C_i$  and  $C_j$ , by assumption, we have:

$$\forall x_i \in C_i, x_j \in C_j, \quad P(x_i, x_j \text{ are NNs}) = p_{ij}$$

where  $p_{ij}$  is constant.

FP edges for a given point are sampled randomly from all non-NN points. For each point,  $n_{FP}$  FP points are randomly selected, where  $n_{FP}$  is a constant defaulting to 20 for PaCMAP. Thus, the expected number of NNs and FPs between  $C_i$  and  $C_j$  are

$$\begin{aligned} \mathbb{E}(\# \text{ of NNs between } C_i, C_j) &= n_i n_j p_{ij} \\ \mathbb{E}(\# \text{ of FPs between } C_i, C_j) &= \frac{2n_i n_j n_{FP}}{n}, \end{aligned} \quad (2)$$

because each point  $i \in C_i$  selects  $\frac{n_j}{n} \cdot n_{FP}$  FP pairs, the total number of points in  $C_i$  sampled from  $C_i$  to  $C_j$  is  $\frac{n_i \cdot n_j \cdot n_{FP}}{n}$  FP pairs. Similarly,  $\frac{n_i \cdot n_j \cdot n_{FP}}{n}$  FP pairs are sampled from all points in  $C_j$  between  $C_i$  and  $C_j$ . Therefore,  $\frac{2n_i n_j \cdot n_{FP}}{n}$  total FP pairs are sampled between these two clusters. Therefore, the ratio between the number of NN edges and the number of FP edges is

$$\frac{\mathbb{E}(\# \text{ of NNs between } C_i, C_j)}{\mathbb{E}(\# \text{ of FPs between } C_i, C_j)} = \frac{n \cdot p_{ij}}{2n_{FP}}.$$

Considering that  $p_{ij}$  is unaffected by  $n$  and  $n_{FP}$ ,  $n_i$  and  $n_j$  are constants, the ratio increases with  $n$ . This result is true for all clusters  $C_i$  and  $C_j$ . Thus, for each pair of clusters, the ratio of NN edges to FP edges grows linearly in  $n$ . This completes the proof.  $\square$

This phenomenon is further illustrated in Figure 3. When DR is applied to the entire MNIST dataset, images of six different digits are grouped into two large clusters, with each

cluster containing three digit classes. However, when DR is applied separately to each of these large clusters, they are successfully separated into distinct clusters, each representing a single digit class. It shows that when the sample size increases, usually more structure is involved in the dataset, which indicates higher complexity and larger number of classes.

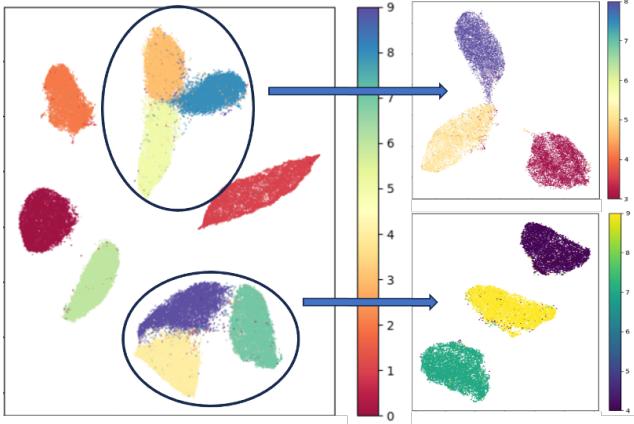


Figure 3: *Left:* PaCMAP embedding on the entire MNIST dataset where six clusters are grouped into two large ones (each with three digit classes). *Right:* PaCMAP embeddings on each of the two groups of three digit classes. The right embeddings work when the left do not because the partial datasets are smaller and thus do not suffer from the problem identified in Insight 2.

## 5 LocalMAP

The insights above suggest a new approach to DR that keeps information *local*, so as to mitigate the problem of Insight 2, and to decrease the impact of false positive edges, avoiding the pitfall of Insight 1. The way LocalMAP handles this is to *replicate a small-scale DR within its large-scale computation*. Specifically, we increase both NN attractive forces and FP repulsive forces locally compared to standard DR approaches, adjusting weights *dynamically* as we learn more about which data points are false positives.

This approach has few downsides, if any. Because it sees more local information, and because it reduces the impact of false positive edges, LocalMAP is able to better capture local structure. It is slightly more computationally expensive than some of the regular DR approaches and faster than others, even for large datasets, as we show in Section 8.4.

### 5.1 LocalMAP Algorithm

As discussed, LocalMAP handles Insight 1 and 2 by increasing local computation. LocalMAP is shown in Algorithm 1 with a detailed version in Appendix A. Major differences from previous DR approaches are:

**LocalMAP Computation 1.** Adjusting the weights for NN edges dynamically and locally according to the low-

dimensional distance during optimization. This is done according to a set of principles listed below.

**LocalMAP Computation 2.** Resampling FP edges to be local. This allows LocalMAP to separate nearby clusters. In practice, local FP edges are randomly selected non-neighbor pairs with distance no larger than the average low-dimension distance among all nearest clusters pairs; we call this the *proximal cluster distance common*  $\bar{d}_{\text{adj}}$ .

Both LocalMAP Computation 1 and LocalMAP Computation 2 apply to the final stage of optimization. LocalMAP uses earlier stages from another DR approach to handle global structure placement, which is sufficient as set up for LocalMAP’s computations.

### 5.2 LocalMAP’s principles for NN edge weighting in LocalMap Computation 1

LocalMAP creates several central aspects for DR. Specifically, these principles are:

1. Increased weights for NNs that are close in low-dimensional space. (These are estimated to be true positive pairs.)
2. Decreased weights for NNs that are far in low-dimensional space. (These are estimated to be false positive pairs.)
3. Avoid extremely large forces, ensuring stable convergence.
4. The weighting function needs to be simple, easy to combine with the NN loss terms, and fast to compute.

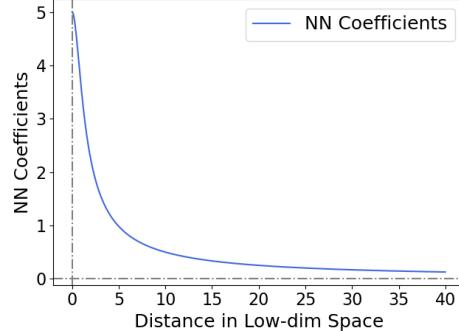


Figure 4: Curve of  $\text{Coefficient}_{\text{NN}}$ .

A natural choice to achieve Principles 1 and 2 and 4 is to use a weighting function that is inversely proportional to the low-dimensional distance. However, this could lead to very large values for small low-dimensional distances, making the convergence unstable, violating Principle 3.

Moreover, to define “close” in Principles 1 and 2, the embedding scale should be considered. We consider the average distance between adjacent clusters in PaCMAP embedding,  $\bar{d}_{\text{adj}}$ , which is approximately 10 based on observations. The midpoint between the two clusters ( $\frac{\bar{d}_{\text{adj}}}{2}$ ) could serve as a threshold to determine whether to increase or reduce the attractive force. The coefficient should be greater than 1 when

---

**Algorithm 1: Implementation of LocalMAP**


---

**Require:**  $\mathbf{X}$  - data matrix,  $\bar{d}_{\text{adj}}$  from Section 5, parameter  $C_{\text{Med}} \sim 10$ .

**Ensure:**

- Initialized low-dimensional embedding  $\mathbf{Y}$ .
- Construct neighbor pairs, mid-near pairs and further pairs according to high-dimensional distance.
- Optimize  $\mathbf{Y}$  using the first two training phases of PaCMAP.
- Optimize  $\mathbf{Y}$  using loss:

$$\text{Loss}(\mathbf{Y}) := \sum_{(i,j): \text{NN}} \frac{\bar{d}_{\text{adj}} \cdot \sqrt{\tilde{d}_{ij}}}{2(C_{\text{Med}} + \tilde{d}_{ij})} + \sum_{(i,l): \text{FP}} \frac{1}{1 + \tilde{d}_{il}},$$

where  $\tilde{d}_{ij} = \|\mathbf{y}_i - \mathbf{y}_j\|^2 + 1$ .

The FP pairs are resampled every 10 iterations to stay *local* so that all FPs satisfies  $\|\mathbf{y}_i - \mathbf{y}_l\| \leq \bar{d}_{\text{adj}}, \forall (i, l) \in \text{FP pairs}$ .  
**return**  $\mathbf{Y}$

---

the attraction force needs to be increased and less than 1 when it needs to be reduced.

To achieve all four principles, we choose a weighting function as follows:

$$\text{Coefficient}_{\text{NN}}(d_{ij}) = \frac{\frac{\bar{d}_{\text{adj}}}{2}}{\sqrt{d_{ij}^2 + 1}} = \frac{\bar{d}_{\text{adj}}}{2\sqrt{\tilde{d}_{ij}}}.$$

Based on the above equation, when  $\frac{\bar{d}_{\text{adj}}}{2} > \sqrt{\tilde{d}_{ij}} \approx d_{ij}$ , the attractive force along the  $(i, j)$  pair increases, and this force would decrease when  $\frac{\bar{d}_{\text{adj}}}{2} < \sqrt{\tilde{d}_{ij}} \approx d_{ij}$ .

Figure 4 shows how Coefficient<sub>NN</sub> changes with the low dimensional distance between pairs of NN. Implementing this by adapting PaCMAP's loss and setting  $C_{\text{Med}}$  to 10 to fix the scale of the embedding, its NN loss term becomes:

$$\begin{aligned} \text{Loss}_{\text{NN}} &= w_{\text{NN}} \cdot \sum_{(i,j): \text{NN}} \frac{\tilde{d}_{ij}}{C_{\text{Med}} + \tilde{d}_{ij}} \cdot \frac{\bar{d}_{\text{adj}}}{2\sqrt{\tilde{d}_{ij}}} \\ &= w_{\text{NN}} \cdot \sum_{(i,j): \text{NN}} \frac{\bar{d}_{\text{adj}} \cdot \sqrt{\tilde{d}_{ij}}}{2(C_{\text{Med}} + \tilde{d}_{ij})}. \end{aligned} \quad (3)$$

As stated in Section 2, DR methods are unsupervised and no common objective function exists; we proved in Theorem 2 that LocalMAP also satisfies the six principles mentioned in Wang et al. (2021).

**Theorem 2.** *LocalMAP's loss function obeys the six principles of Wang et al. (2021) for any choices of  $\bar{d}_{\text{adj}}$ , excluding NNs that have low dimensional distances larger than a threshold with value  $\sqrt{C_{\text{Med}} - 1}$  (i.e., possible false positives), where we set  $C_{\text{Med}} \sim 10$  across datasets to determine the scale of the embedding.*

The proof is given in Appendix C.

## 6 Case Study

Figure 5 shows the results for several DR methods on three datasets. Two of the datasets are handwritten digit datasets (LeCun, Cortes, and Burges 2010; Hull 1994), which means

that there are distinct clusters in high dimensions, but also these datasets are special in that Euclidean distance (which is used for defining the graph for DR) is not the same as the geodesic distance along the data manifold, meaning there will be plenty of false positive edges, similar to those shown in Figure 2. The last dataset is a biological dataset (Kang et al. 2018) that contains multiple cell types that are labeled.

We observe that *LocalMAP does a far better job in separating clusters on all three datasets*. A quantitative result is given by the Silhouette score shown in the figures. The definition of Silhouette score is in Appendix E. LocalMAP's scores are superior, confirming what we see visually in these DR plots. There are 5 additional datasets analyzed in Section 8 and visualized in Appendix H, all with similarly impressive visual results.

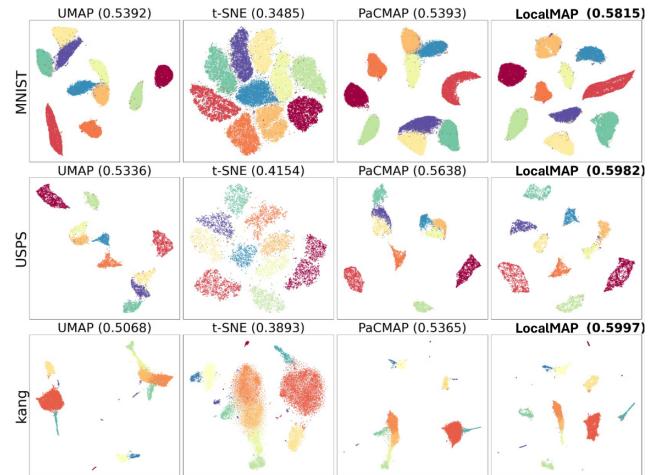


Figure 5: Case study on MNIST (LeCun, Cortes, and Burges 2010), USPS (Hull 1994) and Kang (Kang et al. 2018). The Silhouette scores are shown in parentheses.

## 7 Ablation Study

To understand how each part of the modification contributes to the DR results — an adjustment of NN edge weights and

local FP edge resampling — we conducted an ablation study, as shown in Figure 6. The left plot in the figure displays the embedding generated by LocalMAP with adjusted NN edge weights but without local FP resampling. In this embedding, the clusters have higher density, yet the separation between nearby clusters remains inadequate. The right plot illustrates the embedding generated by LocalMAP with local FP resampling but without adjusted NN edge weights. Although the clusters are more dispersed, the separation between nearby clusters is still insufficient. Therefore, these results indicate that both NN edge weight adjustment and local FP edge resampling are necessary to achieve clear separation between clusters. These observations are clear with MNIST, which is why it is useful to work with this dataset; the same observations persist with other datasets.



Figure 6: Ablation study of LocalMAP on MNIST dataset. **Left:** LocalMAP with adjusted NN edge weights but without local FP resampling. **Right:** LocalMAP with local FP resampling but without adjusted NN edge weights.

## 8 Experiments

### 8.1 Experimental Setup

**Datasets.** Inspired by other DR studies (Tang et al. 2016; McInnes, Healy, and Melville 2018; Amid and Warmuth 2019; Huang et al. 2022), the following datasets are used to evaluate and compare DR methods: MNIST (LeCun, Cortes, and Burges 2010), FMNIST (Xiao, Rasul, and Vollgraf 2017), USPS (Hull 1994), COIL20 (Nene, Nayar, and Murase 1996), 20NG (Lang 1995), Seurat (Stuart et al. 2019), Kang (Kang et al. 2018), Human Cortex (Zhu et al. 2023) and CBMC (Stoeckius et al. 2017).

**Algorithms.** We evaluate LocalMAP in comparison with several other DR techniques: t-SNE (van der Maaten and Hinton 2008), UMAP (McInnes, Healy, and Melville 2018), PaCMAP (Wang et al. 2021), TriMap (Amid and Warmuth 2019), PHATE (Moon et al. 2017), NCVIS (Artemenkov and Panov 2020), h-NNE (Sarfraz et al. 2022), Neg-t-SNE (Damrich et al. 2023), and InfoNC-t-SNE (Damrich et al. 2023). For the t-SNE algorithm, we utilize the openTSNE implementation (Poličar, Stražar, and Zupan 2023).

**Computational Environment.** All experiments were run on a 12 Core Intel(R) Xeon(R) CPU E5640 @ 2.67GHz with

a GPU RTX2080Ti, with memory limit 32G. All experiments were run 10 times for error bar computation.

**Biological Dataset Preprocessing.** For the single-cell datasets, all variables were normalized and log-transformed by the SCANPY package (Wolf, Angerer, and Theis 2018). The top 1000 variance genes within each dataset were selected before applying LocalMAP. For the datasets that have different batches, the ComBat algorithm (Johnson, Li, and Rabinovic 2007) was used to remove batch effects. The detailed dataset description are shown within Appendix D.

### 8.2 Experimental Results: Silhouette Score

Evaluation of DR approaches is challenging because DR methods are unsupervised. There are numerous evaluation methods that characterize different aspects of performance; an overview is given by Huang et al. (2022), though several DR papers propose their own evaluation metrics (e.g., Amid and Warmuth 2019). In this work, we use the silhouette score (Rousseeuw 1987) as our evaluation metric because it captures how clearly the clusters are separated, which no other metric captures. The definition of the sihouette score is in Appendix E.

All the methods are evaluated based on their default hyperparameters. The result in Table 1 shows that LocalMAP separates clusters better than other approaches. This table simply quantifies the high quality clusters we see visually in the LocalMAP plots throughout this paper. We have also shown the performance of UMAP, LargeVis, t-SNE, and PHATE with tuned hyperparameters, and additional results are in Table 6 in Appendix J. *LocalMAP shows a better separation among clusters.*

### 8.3 Experimental Results: Posthoc Classification

DR methods are unsupervised. Here, we consider an evaluation of whether class information, which is not presented to the DR algorithm, is preserved during the process of DR. This is a type of local structure evaluation, but unlike the silhouette score, it does not consider the margins between classes and has several other problems as an evaluation metric, discussed in Appendix F. Essentially, clusters that visually appear merged or are broken up into subclusters (i.e., poor DR plots) can still yield high posthoc classification scores. Table 3 and 7 in Appendix J show the results, which is that LocalMAP achieves similar posthoc classification performance to 11 state-of-the-art DR methods. (But, as discussed, this evaluation measure is problematic.)

### 8.4 Experimental Results: Runtime

Table 2 shows runtimes for several algorithms. LocalMAP is slightly more computationally intensive than other methods because it resamples the graph dynamically, but it is still efficient. Efficiency is typically not as important as DR quality, as evaluated in Section 8.2.

### 8.5 Experimental Results: Robustness and Sensitivity

DR results cannot be trusted if they change under random initialization. Hence, an important characteristic of DR al-

Table 1: Silhouette scores for different Algorithms. **Bold** is best, underline is second best or statistically insignificant from the best. Each row is an algorithm, each column is a dataset.

	MNIST	FMNIST	USPS	COIL20	20NG	Kang	Seurat	Human Cortex	CBMC
<b>PCA</b>	0.02±0.00	-0.03±0.00	0.10±0.00	0.01±0.00	-0.19±0.00	0.12±0.00	-0.06±0.00	-0.08±0.00	-0.11±0.00
<b>t-SNE</b>	0.35±0.00	0.12±0.00	0.42±0.00	0.41±0.00	<u>-0.11±0.00</u>	0.40±0.01	0.22±0.01	0.11±0.01	0.15±0.01
<b>UMAP</b>	0.52±0.01	<b>0.19±0.00</b>	0.53±0.00	<b>0.58±0.01</b>	-0.15±0.01	0.51±0.00	0.30±0.00	0.12±0.02	<u>0.22±0.00</u>
<b>PaCMAP</b>	<u>0.54±0.01</u>	<b>0.19±0.00</b>	<u>0.56±0.00</u>	0.51±0.02	<u>-0.11±0.01</u>	0.53±0.00	<u>0.31±0.00</u>	<u>0.13±0.01</u>	<u>0.22±0.00</u>
<b>LargeVis</b>	0.49±0.05	0.11±0.03	0.41±0.12	0.38±0.01	-0.13±0.01	0.44±0.01	0.25±0.01	0.10±0.02	0.17±0.00
<b>TriMAP</b>	0.41±0.00	<u>0.17±0.00</u>	0.48±0.00	0.47±0.00	-0.13±0.00	<u>0.55±0.00</u>	<b>0.32±0.00</b>	0.07±0.00	0.21±0.00
<b>PHATE</b>	0.26±0.02	0.11±0.01	0.27±0.01	0.33±0.00	-0.21±0.01	0.48±0.02	0.27±0.01	-0.09±0.01	0.06±0.01
<b>HNNE</b>	0.21±0.03	0.06±0.04	0.23±0.00	0.03±0.00	-0.34±0.03	0.39±0.06	-0.00±0.03	-0.09±0.06	0.12±0.05
<b>Neg-t-SNE</b>	0.48±0.00	<b>0.19±0.00</b>	0.48±0.00	0.44±0.01	<u>-0.11±0.00</u>	0.53±0.00	<b>0.32±0.00</b>	0.12±0.00	<b>0.24±0.00</b>
<b>NCVis</b>	0.38±0.02	<b>0.19±0.00</b>	0.44±0.00	0.53±0.00	-0.15±0.00	0.51±0.00	0.27±0.00	0.10±0.00	0.20±0.00
<b>InfoNC-t-SNE</b>	0.33±0.00	0.13±0.00	0.37±0.00	0.43±0.01	<u>-0.11±0.00</u>	0.46±0.00	0.26±0.00	0.10±0.00	0.21±0.00
<b>LocalMAP</b>	<b>0.58±0.00</b>	<b>0.19±0.00</b>	<b>0.60±0.00</b>	<u>0.56±0.01</u>	<b>-0.10±0.00</b>	<b>0.60±0.00</b>	<b>0.32±0.00</b>	<b>0.14±0.00</b>	<u>0.22±0.00</u>

Table 2: Running Time for Different Algorithms. Each row is an algorithm, each column is a dataset.

	MNIST	FMNIST	USPS	COIL20	20NG	Kang	Seurat	Human Cortex	CBMC
<b>PCA</b>	00:01.77	00:01.64	00:00.13	00:02.77	00:00.11	00:00.07	00:00.07	00:00.22	00:00.20
<b>t-SNE</b>	03:02.38	02:36.97	00:35.92	00:10.36	01:10.69	00:40.33	01:05.32	01:05.68	01:49.60
<b>UMAP</b>	00:28.05	00:33.94	00:08.80	00:08.76	00:08.87	00:08.38	00:13.51	00:27.18	00:29.33
<b>PaCMAP</b>	00:52.39	00:34.54	00:04.26	00:03.28	00:08.29	00:05.59	00:13.16	00:18.72	00:39.59
<b>LargeVis</b>	14:51.71	14:02.86	06:45.63	07:09.97	06:44.39	07:12.39	08:15.04	08:31.17	11:18.19
<b>TriMAP</b>	00:53.64	00:48.16	00:06.59	00:01.29	00:13.05	00:13.26	00:19.31	00:26.75	01:03.43
<b>PHATE</b>	04:15.00	01:45.05	00:11.40	00:05.15	00:19.44	00:20.12	00:42.26	01:25.71	06:26.80
<b>HNNE</b>	00:10.05	00:06.41	00:00.60	00:02.35	00:02.19	00:01.33	00:04.21	00:03.02	00:04.27
<b>Neg-t-SNE</b>	01:02.65	01:10.92	00:12.62	00:03.86	00:23.06	00:18.32	00:28.25	00:42.77	00:56.06
<b>NCVis</b>	02:36.92	01:39.70	00:12.92	00:14.16	00:26.69	00:31.72	00:42.20	01:03.09	02:16.05
<b>InfoNC-t-SNE</b>	01:06.19	01:01.58	00:14.30	00:05.63	00:25.31	00:18.85	00:36.02	00:46.74	01:04.94
<b>LocalMAP</b>	01:47.35	00:47.51	00:06.23	00:06.77	00:13.28	00:07.63	00:20.17	00:29.02	01:12.35

gorithms is that they produce consistent results with different initial conditions. Figure 17 in Appendix I shows the result of LocalMAP over several runs, showing that it is capable of consistently producing correct clustering results, whereas other methods do not. In fact, *there are no runs of any other methods that distinctly separate the 10 clusters that LocalMAP finds every time*.

## 9 Discussion

While in the past, DR methods aimed to maintain the local and global structure inherent in high-dimensional data, these methods trusted its underlying graph structure, usually derived from an untrustworthy distance metric. LocalMAP does not do this, instead it dynamically (during run time) discovers what parts of the graph are untrustworthy, and what parts of the graph are not sampled well enough to be able to tell clusters apart. This is why its results are visually and quantitatively better than other DR methods – it essentially

cleans up the data while it is running.

One direction for future work would be to combine the insights of LocalMAP with new parametric approaches to DR that have just begun to yield successful results (Huang, Wang, and Rudin 2024).

Our work could have substantial societal impact, for instance, if it is able to find clusters of patients that have different immune system properties (Semenova et al. 2024; Falconelli et al. 2023). Our experiments indicate that LocalMAP has a higher chance of accomplishing this than past DR approaches.

## Acknowledgments

We acknowledge funding from the National Science Foundation under grants DGE-2022040, CMMI-2323978, and IIS-2130250 and the National Institutes of Health (NIH) under grant R01-DA054994 and other transactions award 1OT2OD032701-01.

## References

- Amezquita, R. A.; Lun, A. T.; Becht, E.; Carey, V. J.; Carpp, L. N.; Geistlinger, L.; Marini, F.; Rue-Albrecht, K.; Risso, D.; Soneson, C.; et al. 2020. Orchestrating Single-Cell Analysis with Bioconductor. *Nature Methods*, 17(2): 137–145.
- Amid, E.; and Warmuth, M. K. 2019. TriMAP: Large-scale Dimensionality Reduction Using Triplets. *arXiv:1910.00204*.
- Artemenkov, A.; and Panov, M. 2020. NCVIS: Noise Contrastive Approach for Scalable Visualization. In *Proceedings of The Web Conference 2020*, 2941–2947. New York, NY, USA: Association for Computing Machinery.
- Atitey, K.; Motsinger-Reif, A. A.; and Anchang, B. 2024. Model-based evaluation of spatiotemporal data reduction methods with unknown ground truth through optimal visualization and interpretability metrics. *Briefings in Bioinformatics*, 25(1): bbad455.
- Becht, E.; McInnes, L.; Healy, J.; Dutertre, C.-A.; Kwok, I. W.; Ng, L. G.; Ginhoux, F.; and Newell, E. W. 2019. Dimensionality reduction for visualizing single-cell data using UMAP. *Nature Biotechnology*, 37(1): 38–44.
- Belkin, M.; and Niyogi, P. 2001. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Advances in Neural Information Processing Systems*, volume 14, 585–591. MIT Press.
- Belkina, A. C.; Ciccolella, C. O.; Anno, R.; Halpert, R.; Spidlen, J.; and Snyder-Cappione, J. E. 2019. Automated optimized parameters for T-distributed stochastic neighbor embedding improve visualization and analysis of large datasets. *Nature Communications*, 10(5415).
- Böhm, J. N.; Berens, P.; and Kobak, D. 2020. A Unifying Perspective on Neighbor Embeddings along the Attraction-Repulsion Spectrum. *arXiv:2007.08902*.
- Böhm, J. N.; Berens, P.; and Kobak, D. 2023. Unsupervised visualization of image datasets using contrastive learning. In *International Conference on Learning Representations*.
- Cao, J.; Spielmann, M.; Qiu, X.; Huang, X.; Ibrahim, D. M.; Hill, A. J.; Zhang, F.; Mundlos, S.; Christiansen, L.; Steemers, F. J.; et al. 2019. The single-cell transcriptional landscape of mammalian organogenesis. *Nature*, 566(7745): 496–502.
- Cao, Y.; and Wang, L. 2017. Automatic selection of t-SNE Perplexity. *arXiv:1708.03229*.
- Chen, P.; Liao, B.; Chen, G.; and Zhang, S. 2019. Understanding and Utilizing Deep Neural Networks Trained with Noisy Labels. In *International Conference on Machine Learning*.
- Damrich, S.; Böhm, J. N.; Hamprecht, F. A.; and Kobak, D. 2023. From t-SNE to UMAP with contrastive learning. In *International Conference on Learning Representations*.
- Dries, R.; Zhu, Q.; Dong, R.; Eng, C.-H. L.; Li, H.; Liu, K.; Fu, Y.; Zhao, T.; Sarkar, A.; Bao, F.; et al. 2021. Giotto: a toolbox for integrative analysis and visualization of spatial expression data. *Genome Biology*, 22: 1–31.
- Falcinelli, S. D.; Cooper-Volkheimer, A. D.; Semenova, L.; Wu, E.; Richardson, A.; Ashokkumar, M.; Margolis, D. M.; Archin, N. M.; Rudin, C. D.; Murdoch, D.; and Browne, E. P. 2023. Impact of Cannabis Use on Immune Cell Populations and the Viral Reservoir in People With HIV on Suppressive Antiretroviral Therapy. *The Journal of Infectious Diseases*, jiad364.
- Gove, R.; Cadalzo, L.; Leiby, N.; Singer, J. M.; and Zaitzeff, A. 2022. New guidance for using t-SNE: Alternative defaults, hyperparameter selection automation, and comparative evaluation. *Visual Informatics*, 6(2): 87–97.
- Han, B.; Yao, Q.; Yu, X.; Niu, G.; Xu, M.; Hu, W.; Tsang, I. W.; and Sugiyama, M. 2018. Co-teaching: robust training of deep neural networks with extremely noisy labels. In *Neural Information Processing Systems*, 8536–8546.
- Huang, H.; Wang, Y.; and Rudin, C. 2024. Navigating the Effect of Parametrization for Dimensionality Reduction. In *Neural Information Processing Systems*.
- Huang, H.; Wang, Y.; Rudin, C.; and Browne, E. P. 2022. Towards a comprehensive evaluation of dimension reduction methods for transcriptomic data visualization. *Communications Biology*, 5(1): 719.
- Hull, J. J. 1994. A database for handwritten text recognition research. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(5): 550–554.
- Johnson, W. E.; Li, C.; and Rabinovic, A. 2007. Adjusting batch effects in microarray expression data using empirical Bayes methods. *Biostatistics*, 8(1): 118–127.
- Kang, H. M.; Subramaniam, M.; Targ, S.; Nguyen, M.; Maliskova, L.; McCarthy, E.; Wan, E.; Wong, S.; Byrnes, L.; Lanata, C. M.; et al. 2018. Multiplexed droplet single-cell RNA-sequencing using natural genetic variation. *Nature Biotechnology*, 36(1): 89.
- Kobak, D.; and Linderman, G. C. 2021. Initialization is critical for preserving global data structure in both t-SNE and UMAP. *Nature Biotechnology*, 39(2): 156–157.
- Kock, K. H.; Tan, L. M.; Han, K. Y.; Ando, Y.; Jevapatarakul, D.; Chatterjee, A.; Lin, Q. X. X.; Buyamin, E. V.; Sonthalia, R.; Rajagopalan, D.; et al. 2024. Single-cell analysis of human diversity in circulating immune cells. *bioRxiv*, 2024–06.
- Lang, K. 1995. Newsweeder: Learning to filter netnews. In *International Conference on Machine Learning*, 331–339.
- LeCun, Y.; Cortes, C.; and Burges, C. 2010. MNIST handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2.
- Lee, D. D.; and Seung, H. S. 1999. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755): 788–791.
- Li, J.; Socher, R.; and Hoi, S. C. 2020. DivideMix: Learning with Noisy Labels as Semi-supervised Learning. In *International Conference on Learning Representations*.
- Martin, R. D.; Yohai, V. J.; Maronna, R. A.; and Salibián-Barrera, M. 2018. *Robust Statistics: Theory and Methods (with R)*, 2nd Edition. Wiley.
- McInnes, L.; Healy, J.; and Melville, J. 2018. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv:1802.03426*.
- Moon, K. R.; van Dijk, D.; Wang, Z.; Chen, W.; Hirn, M. J.; Coifman, R. R.; Ivanova, N. B.; Wolf, G.; and Krishnaswamy,

- S. 2017. PHATE: a dimensionality reduction method for visualizing trajectory structures in high-dimensional biological data. *BioRxiv*, 120378.
- 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. 2019. Visualizing structure and transitions in high-dimensional biological data. *Nature Biotechnology*, 37(12): 1482–1492.
- Mu, J.; Bhat, S.; and Viswanath, P. 2017. All-but-the-Top: Simple and Effective Postprocessing for Word Representations. *arXiv:1702.01417*.
- Nene, S. A.; Nayar, S. K.; and Murase, H. 1996. Columbia Object Image Library (COIL-20). Technical report, Technical Report CUCS-005-96.
- Nguyen, L. H.; and Holmes, S. 2019. Ten quick tips for effective dimensionality reduction. *PLoS Computational Biology*, 15(6).
- Northcutt, C. G.; Athalye, A.; and Mueller, J. 2024. Pervasive label errors in test sets destabilize machine learning benchmarks. *Neural Information Processing Systems*.
- Pearson, K. 1901. On Lines and Planes of Closest Fit to Systems of Points in Space. *Philosophical Magazine*, 2(11): 559–572.
- Perez, R. K.; Gordon, M. G.; Subramaniam, M.; Kim, M. C.; Hartoularos, G. C.; Targ, S.; Sun, Y.; Ogorodnikov, A.; Bueno, R.; Lu, A.; et al. 2022. Single-cell RNA-seq reveals cell type-specific molecular and genetic associations to lupus. *Science*, 376(6589): eabf1970.
- Poličar, P. G.; Stražar, M.; and Zupan, B. 2023. Embedding to reference t-SNE space addresses batch effects in single-cell classification. *Machine Learning*, 112(2): 721–740.
- Raunak, V.; Gupta, V.; and Metze, F. 2019. Effective Dimensionality Reduction for Word Embeddings. In *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*, 235–243.
- Rousseeuw, P. J. 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20: 53–65.
- Roweis, S. T.; and Saul, L. K. 2000. Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science*, 290(5500): 2323–2326.
- Sarfraz, S.; Koulakis, M.; Seibold, C.; and Stiefelhagen, R. 2022. Hierarchical nearest neighbor graph embedding for efficient dimensionality reduction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 336–345.
- Semenova, L.; Wang, Y.; Falcinelli, S.; Archin, N.; Cooper-Volkheimer, A. D.; Margolis, D. M.; Goonetilleke, N.; Murdoch, D. M.; Rudin, C. D.; and Browne, E. P. 2024. Machine learning approaches identify immunologic signatures of total and intact HIV DNA during long-term antiretroviral therapy. *eLife*, 13: RP94899.
- Stoeckius, M.; Hafemeister, C.; Stephenson, W.; Houck-Loomis, B.; Chattopadhyay, P. K.; Swerdlow, H.; Satija, R.; and Smibert, P. 2017. Simultaneous epitope and transcriptome measurement in single cells. *Nature Methods*, 14(9): 865–868.
- Stuart, T.; Butler, A.; Hoffman, P.; Hafemeister, C.; Papalexis, E.; Mauck III, W. M.; Hao, Y.; Stoeckius, M.; Smibert, P.; and Satija, R. 2019. Comprehensive integration of single-cell data. *Cell*, 177(7): 1888–1902.
- Tang, J.; Liu, J.; Zhang, M.; and Mei, Q. 2016. Visualizing large-scale and high-dimensional data. In *Proceedings of the 25th International Conference on the World Wide Web*, 287–297.
- Tenenbaum, J. B.; de Silva, V.; and Langford, J. C. 2000. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, 290(5500): 2319–2323.
- Torgerson, W. 1952. Multidimensional scaling: I Theory and Method. *Psychometrika*, 17(4): 401–419.
- Van Assel, H.; Espinasse, T.; Chiquet, J.; and Picard, F. 2022. A probabilistic graph coupling view of dimension reduction. *Advances in Neural Information Processing Systems*, 35: 10696–10708.
- van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9: 2579–2605.
- Wang, Y.; Huang, H.; Rudin, C.; and 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.
- Wattenberg, M.; Viégas, F.; and Johnson, I. 2016. How to use t-SNE effectively. *Distill*, 1(10): e2.
- Wolf, F. A.; Angerer, P.; and Theis, F. J. 2018. SCANPY: large-scale single-cell gene expression data analysis. *Genome Biology*, 19: 1–5.
- Xiao, H.; Rasul, K.; and Vollgraf, R. 2017. Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms. *arXiv:cs.LG/1708.07747*.
- Zhu, K.; Bendl, J.; Rahman, S.; Vicari, J. M.; Coleman, C.; Clarence, T.; Latouche, O.; Tsankova, N. M.; Li, A.; Brennan, K. J.; et al. 2023. Multi-omic profiling of the developing human cerebral cortex at the single-cell level. *Science Advances*, 9(41): eadg3754.
- Zu, X.; and Tao, Q. 2022. SpaceMAP: Visualizing High-Dimensional Data by Space Expansion. In *International Conference on Machine Learning*, 27707–27723.

## A Detailed LocalMAP Algorithm

---

Algorithm 1: Implementation of LocalMAP

---

**Require:**  $\mathbf{X}$  - data matrix,  $\bar{d}_{\text{adj}}$  from Section 5, parameter  $C_{Med} \sim 10$ .

**Ensure:**

- $\mathbf{Y}$  - low-dimensional data matrix. Initialize  $\mathbf{Y}$  with PCA or random initialization.
- Construct neighbor pairs, mid-near pairs and further pairs according to high-dimensional distance.
- Optimize the low-dimensional embedding  $\mathbf{Y}$  using the first two training phases of PaCMAF.
- For the last training phase, optimize the low-dimensional embedding  $\mathbf{Y}$  using the loss:

$$\text{Loss}(\mathbf{Y}) := \sum_{(i,j): \text{NN}} \frac{\bar{d}_{\text{adj}} \cdot \sqrt{\tilde{d}_{ij}}}{2(C_{Med} + \tilde{d}_{ij})} + \sum_{(i,l): \text{FP}} \frac{1}{1 + \tilde{d}_{il}},$$

where  $\tilde{d}_{ij} = \|\mathbf{y}_i - \mathbf{y}_j\|^2 + 1$ . We use the Adam optimizer, and the FP pairs are resampled every 10 iterations to stay *local* throughout the computation, so that all FPs have low-dimensional distance within  $\bar{d}_{\text{adj}}$  if possible, i.e.,  $\|\mathbf{y}_i - \mathbf{y}_l\| \leq \bar{d}_{\text{adj}}$  for all  $(i, l)$  FP pairs. (We impose a limit of 20 maximum sampling attempts for the FPs due to computational considerations.)

**return**  $\mathbf{Y}$

---

## B Detailed proof of Theorem 1

*Proof.* Considering a dataset with  $n$  data points distributed across  $m$  clusters  $C_1, C_2, \dots, C_m$ , where each cluster  $C_i$  contains  $n_i$  data points ( $n_1 + n_2 + \dots + n_m = n$ ). For any two clusters  $C_i$  and  $C_j$ , by assumption, we have:

$$\forall x_i \in C_i, x_j \in C_j, \quad P(x_i, x_j \text{ are NNs}) = p_{ij}$$

where  $p_{ij}$  is constant.

FP edges for a given point are sampled randomly from all non-NN points. For each point,  $n_{FP}$  FP points are randomly selected, where  $n_{FP}$  is a constant defaulting to 20 for PaCMAP. Thus, the expected number of NNs and FPs between  $C_i$  and  $C_j$  are

$$\begin{aligned} \mathbb{E}(\# \text{ of NNs between } C_i, C_j) &= \sum_{x_i \in C_i, x_j \in C_j} P(x_i, x_j \text{ are NNs}) = \sum_{x_i \in C_i, x_j \in C_j} p_{ij} = n_i n_j p_{ij} \\ \mathbb{E}(\# \text{ of FPs between } C_i, C_j) &= \sum_{x_i \in C_i} \frac{n_j}{n} \cdot n_{FP} + \sum_{x_j \in C_j} \frac{n_i}{n} \cdot n_{FP} = \frac{2n_i n_j n_{FP}}{n}, \end{aligned}$$

because each point  $i \in C_i$  selects  $\frac{n_j}{n} \cdot n_{FP}$  FP pairs, the total number of points in  $C_i$  sampled from  $C_i$  to  $C_j$  is  $\frac{n_i \cdot n_j \cdot n_{FP}}{n}$  FP pairs. Similarly,  $\frac{n_i \cdot n_j \cdot n_{FP}}{n}$  FP pairs are sampled from all points in  $C_j$  between  $C_i$  and  $C_j$ . Therefore,  $\frac{2n_i n_j n_{FP}}{n}$  total FP pairs are sampled between these two clusters. Therefore, the ratio between the number of NN edges and the number of FP edges is

$$\frac{\mathbb{E}(\# \text{ of NNs between } C_i, C_j)}{\mathbb{E}(\# \text{ of FPs between } C_i, C_j)} = \frac{n_i n_j p_{ij}}{2n_i n_j n_{FP}/n} = \frac{n \cdot p_{ij}}{2n_{FP}}.$$

Considering that  $p_{ij}$  is unaffected by  $n$  and  $n_{FP}$ ,  $n_i$  and  $n_j$  are constants, the ratio increases with  $n$ . This result is true for all clusters  $C_i$  and  $C_j$ . Thus, for each pair of clusters, the ratio of NN edges to FP edges grows linearly in  $n$ . This completes the proof.  $\square$

## C Proof of Theorem 2: six principles for DR loss functions

Here we check the conditions identified by (Wang et al. 2021) for high-quality DR loss functions. Consider a triplet  $(i, j, k)$  where  $i$  and  $j$  are high-dimensional neighbors that should be attracted to each other, and  $i$  and  $k$  are further points in the high-dimension that should be repulsed from each other. How forces along pairs  $(i, j)$  and  $(i, k)$  should be affected by the distance between these pairs of points  $d_{ij}$  and  $d_{ik}$  are defined by the six principles.

LocalMAP's loss in the early stages follows these principles automatically. For its last stage, the loss function follows the principles for NNs that are close in low-dimensional space (which indicates they are true positive pairs) with distance smaller than  $\sqrt{C_{Med} - 1} = 3$ . For NNs beyond that, we do not want them to obey the principles because they are probably false positives.

According to Proposition 1 of (Wang et al. 2021), for loss functions of the form:

$$Loss = \sum_{ij} Loss_{attractive}(d_{ij}) + \sum_{ik} Loss_{repulsive}(d_{ik}),$$

where its derivatives are:

$$f(d_{ij}) := \frac{\partial Loss_{attractive}(d_{ij})}{\partial d_{ij}}, \quad g(d_{ik}) := -\frac{\partial Loss_{repulsive}(d_{ik})}{\partial d_{ik}},$$

each of the six principles is obeyed under the following conditions of the loss function's derivatives:

1. The functions  $f(d_{ij})$  and  $g(d_{ik})$  are non-negative and unimodal.
2.  $\lim_{d_{ij} \rightarrow 0} f(d_{ij}) = \lim_{d_{ij} \rightarrow \infty} f(d_{ij}) = 0, \lim_{d_{ik} \rightarrow 0} g(d_{ik}) = \lim_{d_{ik} \rightarrow \infty} g(d_{ik}) = 0$ .

For LocalMAP, the loss function for earlier stages is adopted from PaCMAP and thus obeys the principles. For LocalMAP's last phase, the loss function is:

$$Loss_{attractive}(d_{ij}) = \frac{\bar{d}_{adj} \cdot \sqrt{\tilde{d}_{ij}}}{2(C_{Med} + \tilde{d}_{ij})}, \quad \tilde{d}_{ij} = d_{ij}^2 + 1$$

and thus

$$f(d_{ij}) = \frac{\partial Loss_{attractive}(d_{ij})}{\partial d_{ij}} = \frac{\bar{d}_{adj} \cdot d_{ij}(C_{Med} - d_{ij}^2 - 1)}{2(d_{ij}^2 + 1)^{\frac{1}{2}}(C_{Med} + d_{ij}^2 + 1)} \geq 0 \quad \text{when } 0 \leq d_{ij} \leq \sqrt{C_{Med} - 1}.$$

To show  $f(d_{ij})$  is unimodal:

$$\frac{\partial f(d_{ij})}{\partial d_{ij}} = \frac{\bar{d}_{adj}(d_{ij}^2 + 1)^{-\frac{1}{2}}(C_{Med} + d_{ij}^2 + 1)(2d_{ij}^6 + 3d_{ij}^4 - 6C_{Med}d_{ij}^4 - 6C_{Med}d_{ij}^2 + C_{Med}^2 - 1)}{2(d_{ij}^2 + 1)(C_{Med} + d_{ij}^2 + 1)^4}.$$

Since  $\bar{d}_{adj}(d_{ij}^2 + 1)^{-\frac{1}{2}}(C_{Med} + d_{ij}^2 + 1)$  and  $(d_{ij}^2 + 1)(C_{Med} + d_{ij}^2 + 1)^4$  are always positive given  $\bar{d}_{adj}$  is positive, the sign of  $\frac{\partial f(d_{ij})}{\partial d_{ij}}$  depends on  $\Delta = 2d_{ij}^6 + 3d_{ij}^4 - 6C_{Med}d_{ij}^4 - 6C_{Med}d_{ij}^2 + C_{Med}^2 - 1$ .

Considering  $\Delta$  is a six-degree polynomial of  $d_{ij}$ ,  $\Delta = 0$  has 6 roots in the complex field. Let  $t = d_{ij}^2$ , then

$$\Delta = 2t^3 + 3t^2 - 6C_{Med}t^2 - 6C_{Med}t + C_{Med}^2 - 1 = 0.$$

It is clear that  $\lim_{t \rightarrow -\infty} \Delta = -\infty, \lim_{t \rightarrow \infty} \Delta = \infty$ . When  $t = 0, \Delta = C_{Med} - 1 > 0$ , when  $t = C_{Med} - 1, \Delta < 0$ . Then  $t$  has a root in each of  $(-\infty, 0), (0, C_{Med} - 1)$  and  $(C_{Med} - 1, \infty)$ , where each root of  $t$  corresponds to two roots of  $d_{ij}$ .

For the one negative root of  $t$ , it corresponds to two complex root of  $d_{ij}$ . For the two roots of  $t$  in  $(0, C_{Med} - 1)$  and  $(C_{Med} - 1, \infty)$ , each corresponds to one positive root of  $d_{ij}$  and one negative root of  $d_{ij}$ . Therefore,  $d_{ij}$  has only one root of  $\Delta$  in  $(0, \sqrt{C_{Med} - 1})$ , and  $\frac{\partial f(d_{ij})}{\partial d_{ij}}$  is first positive and then negative in  $(0, \sqrt{C_{Med} - 1})$ , which indicates that  $d_{ij}$  first increases and then decreases in  $(0, \sqrt{C_{Med} - 1})$  and thus is unimodal in  $(0, \sqrt{C_{Med} - 1})$ .

LocalMAP's loss for FP pairs has an analogous proof, so we are done.  $\square$

## D Data Description

Detailed descriptions of the data set are shown as follows:

Dataset	# of samples	# of dimensions
MNIST	70,000	784
FMNIST	70,000	784
USPS	9,298	256
COIL20	1,440	16384
20NG	18,846	100
Kang	13,999	1000
Seurat	30,672	1000
Human Cortex	43,349	1000
CBMC	67,686	1000

## E Silhouette Score

The Silhouette score was originally used to evaluate the quality of clusters in clustering analysis where the ground truth labels are not present. In this paper, we use the Silhouette score to evaluate cluster quality from an unsupervised algorithm using ground truth labels from supervised data by substituting the clusters generated from clustering algorithms to ground truth class labels. In this case, it measures the embedding's within-class cohesion (measured by a point's average distance to other points in the same class) against between-class separation (measured by a point's minimum average distance to points of different classes). It is calculated as follows:

1. For each data point  $i$  in a dataset: Calculate average distance  $a_i$  between  $i$  and all other data points within the same class  $C_i$ :  $a_i = \sum_{j \in C_i, j \neq i} d(i, j) / (|C_i| - 1)$  where  $d(i, j)$  is the distance between data points  $i$  and  $j$ . Then calculate the minimum average distance  $b_i$  of  $i$  to all data points in a different class  $C_j$ :  $b_i = \min_{k \neq i} \sum_{k \in C_k} d(i, k) / |C_k|$ .
2. Calculate the silhouette score  $S_i$  for data point  $i$ :  $S_i = \frac{b_i - a_i}{\max(a_i, b_i)}$ .
3. The overall silhouette score  $S$  for the entire dataset is the average of the silhouette scores for all data points:  $S = \frac{\sum_i S_i}{N}$  where  $N$  is the number of data points in the dataset.

## F Posthoc Classification

Tables 3 show posthoc classification results using SVM. Here, SVM is optimized globally. Posthoc classification aims to determine whether class labels are maintained during DR projection even though labels are not used during the process of DR. LocalMAP’s performance is comparable with other local DR methods with respect to these scores.

There are several problems with using posthoc classification as a performance metric. One problem is that its results are not consistent between classification methods, as we can see from these tables. It also does not handle global structure at all (Huang et al. 2022), meaning that one true cluster could be broken into many subclusters by DR (which would be a poor DR result) and the posthoc classification score could be unchanged. Posthoc classification does not measure margins between clusters. Thus, two true clusters that appear stuck together visually (as a fault of DR) may receive the same posthoc classification score as if they were well separated.

Table 3: SVM Score for Different Algorithms, **Bold** is best, underline is not significantly different from best (with only 1% difference). Each row is an algorithm, each column is a dataset.

	MNIST	FMNIST	USPS	COIL20	20NG	Kang	Seurat	Human Cortex	CBMC
PCA	0.47±0.00	0.55±0.00	0.56±0.00	0.66±0.00	0.15±0.00	0.73±0.00	0.46±0.00	0.57±0.00	0.44±0.00
t-SNE	<b>0.97±0.00</b>	0.74±0.00	<b>0.96±0.00</b>	<b>0.85±0.01</b>	0.45±0.01	<u>0.95±0.00</u>	0.84±0.00	<b>0.82±0.00</b>	0.82±0.00
UMAP	<b>0.97±0.00</b>	0.74±0.01	<u>0.95±0.00</u>	0.82±0.01	0.44±0.01	<u>0.95±0.00</u>	0.83±0.00	0.81±0.00	<u>0.82±0.00</u>
PaCMAP	<b>0.97±0.00</b>	<u>0.74±0.00</u>	<u>0.95±0.00</u>	0.83±0.01	<u>0.46±0.01</u>	<u>0.95±0.00</u>	<b>0.85±0.00</b>	<u>0.81±0.00</u>	<b>0.83±0.00</b>
LargeVis	0.96±0.00	<u>0.74±0.01</u>	0.92±0.06	0.80±0.02	<b>0.47±0.00</b>	<u>0.95±0.00</u>	0.84±0.00	<b>0.82±0.00</b>	0.82±0.00
TriMAP	0.96±0.00	0.73±0.00	<u>0.95±0.00</u>	0.77±0.01	0.42±0.01	<u>0.95±0.00</u>	0.84±0.00	0.79±0.00	<u>0.82±0.00</u>
PHATE	0.86±0.02	0.66±0.01	0.86±0.01	<b>0.84±0.00</b>	0.33±0.01	0.92±0.00	0.77±0.00	0.70±0.01	0.72±0.01
HNNE	0.84±0.03	0.68±0.01	0.82±0.00	0.63±0.00	0.24±0.05	0.90±0.01	0.74±0.01	0.68±0.03	0.73±0.04
Neg-t-SNE	0.96±0.00	0.74±0.00	0.93±0.00	0.81±0.01	0.43±0.01	0.95±0.00	0.84±0.00	0.81±0.00	0.82±0.00
NCVis	0.94±0.01	0.73±0.00	0.92±0.00	0.79±0.00	0.36±0.01	0.94±0.00	0.83±0.00	<b>0.82±0.00</b>	<u>0.82±0.00</u>
InfoNC-t-SNE	0.96±0.00	<u>0.74±0.00</u>	0.93±0.00	0.82±0.01	0.42±0.00	<u>0.95±0.00</u>	<b>0.85±0.00</b>	<u>0.81±0.00</u>	<b>0.83±0.00</b>
LocalMAP	<b>0.97±0.00</b>	<b>0.75±0.00</b>	<b>0.96±0.00</b>	<u>0.83±0.01</u>	<u>0.46±0.01</u>	<b>0.96±0.00</b>	0.84±0.00	0.81±0.00	0.82±0.00

## G The relationship between run time and the number of samples

Figure 7 shows the relationship between the number of samples and the run time.

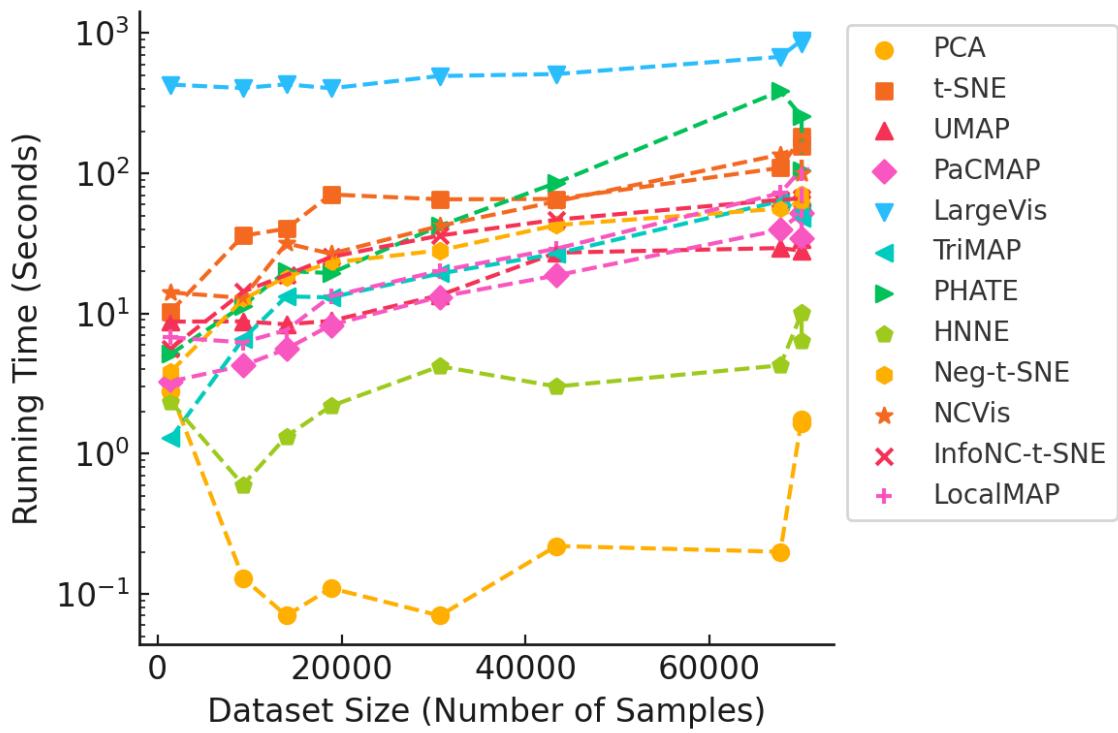


Figure 7: The relationship between the number of samples and the run time (log-scaled in seconds).

## H Additional Visualization for Datasets

We show how different DR methods perform on different datasets in Figures 8 to 16.

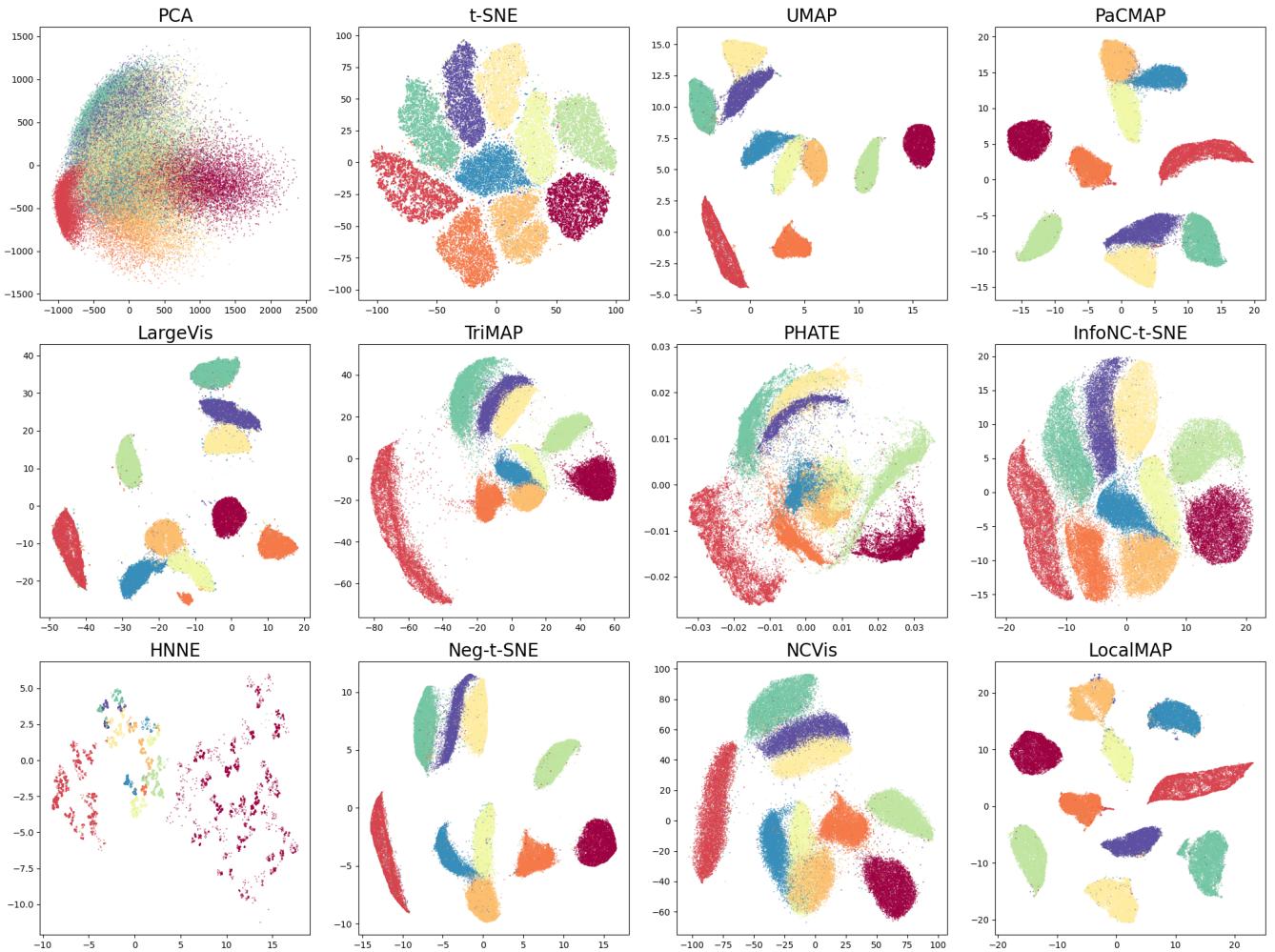


Figure 8: Different DR embeddings for MNIST (LeCun, Cortes, and Burges 2010)

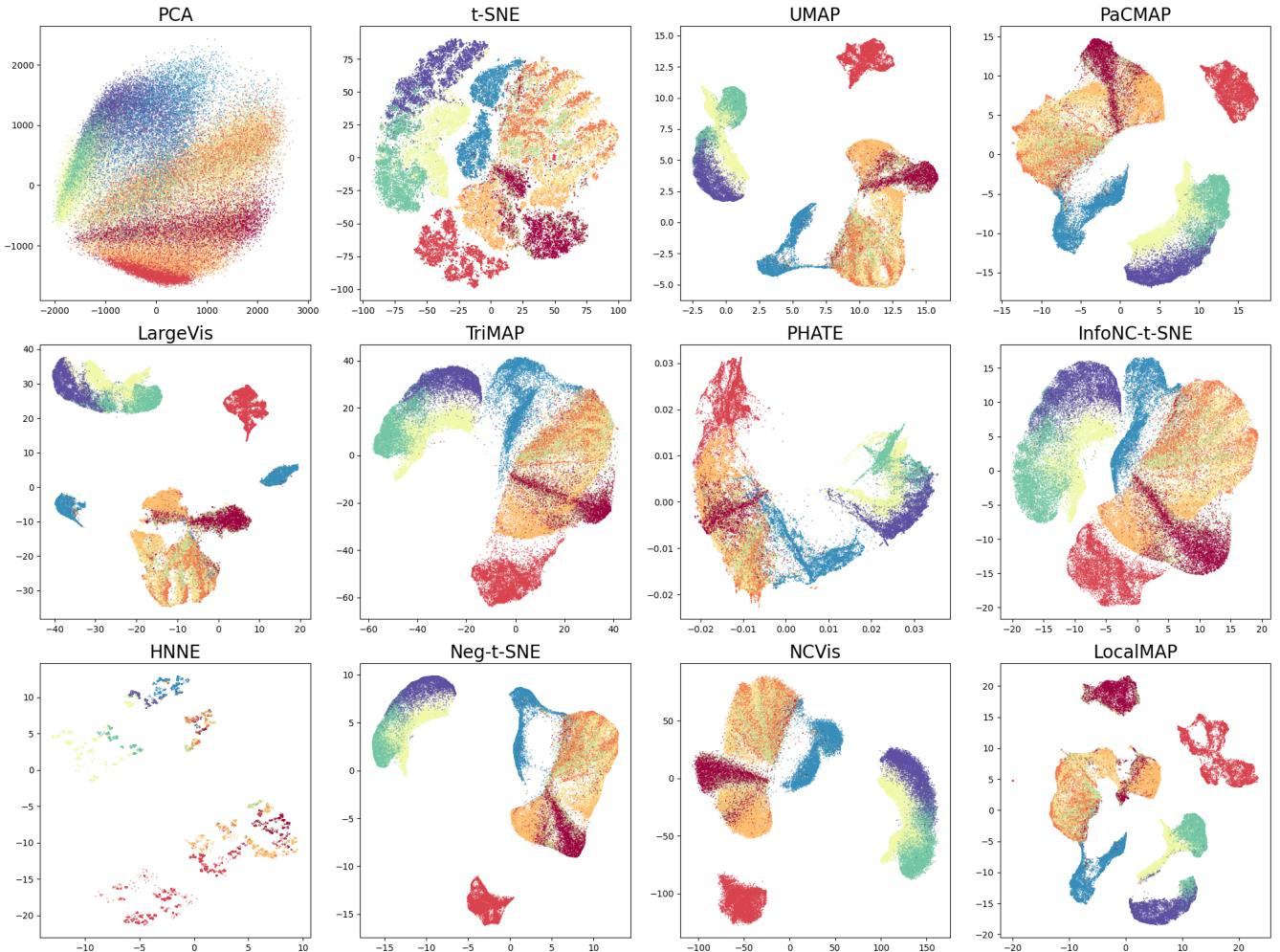


Figure 9: Different DR embeddings for FMNIST (Xiao, Rasul, and Vollgraf 2017)

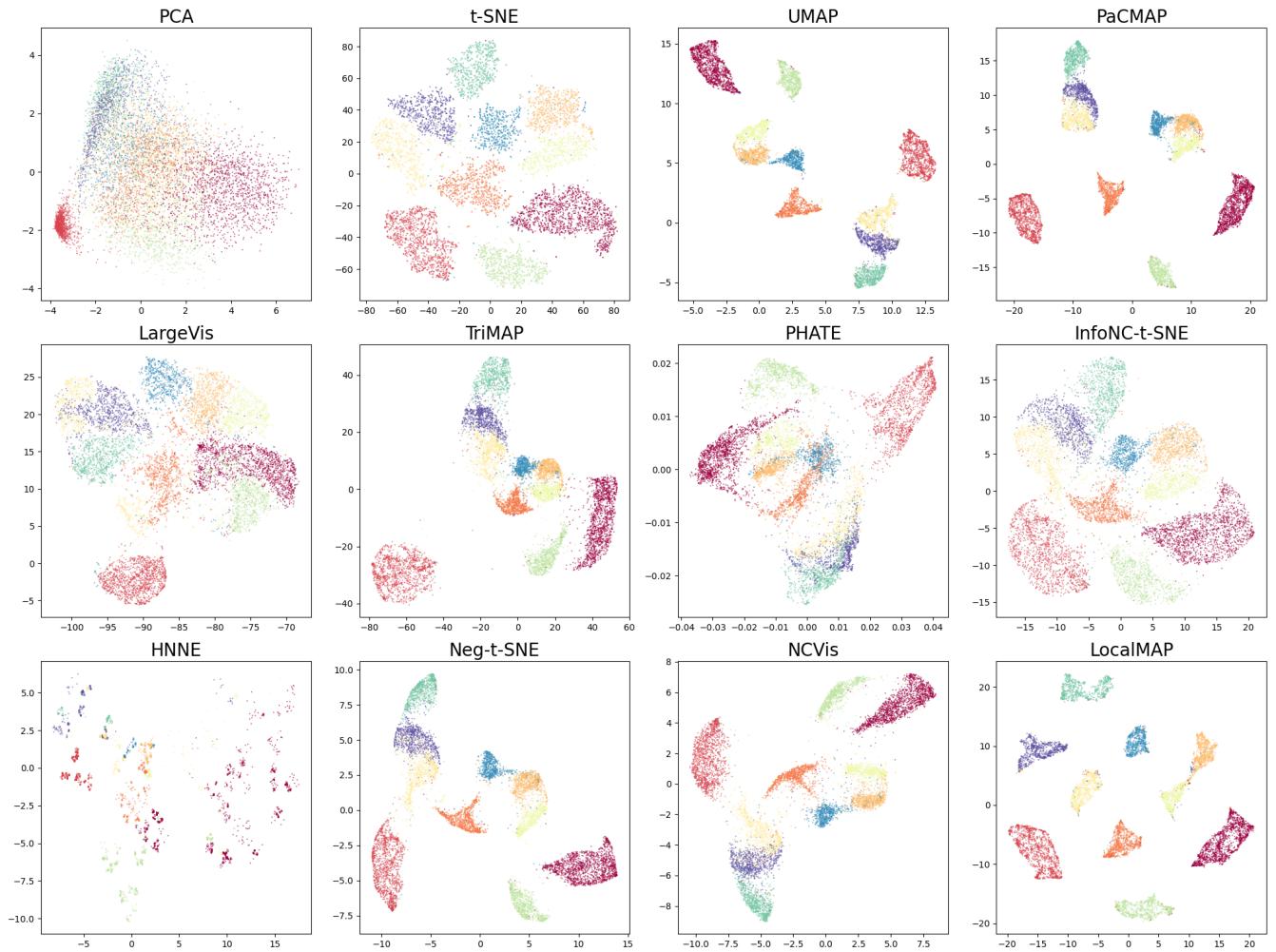


Figure 10: Different DR embeddings for USPS (Hull 1994)

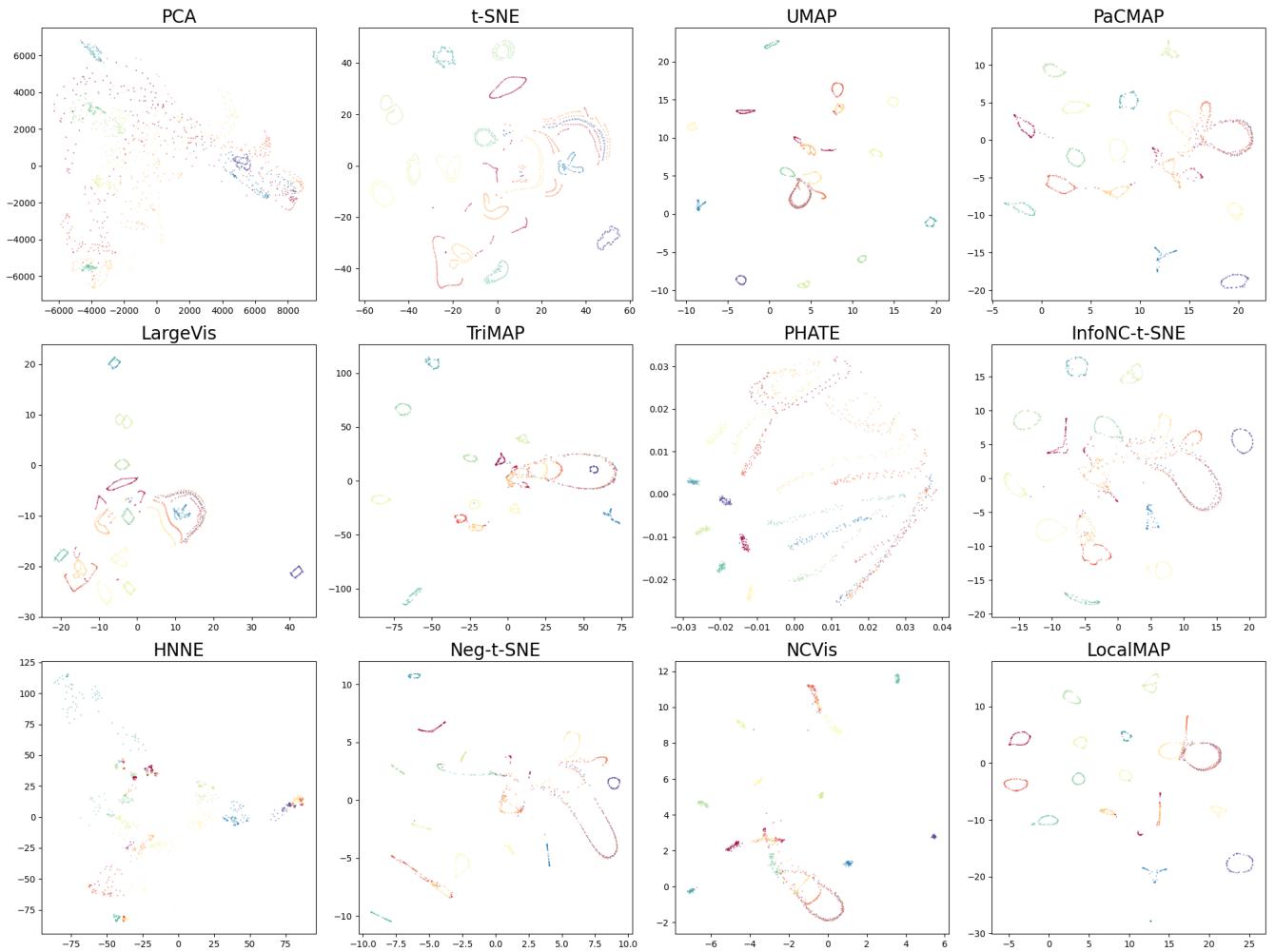


Figure 11: Different DR embeddings for COIL20 (Nene, Nayar, and Murase 1996)

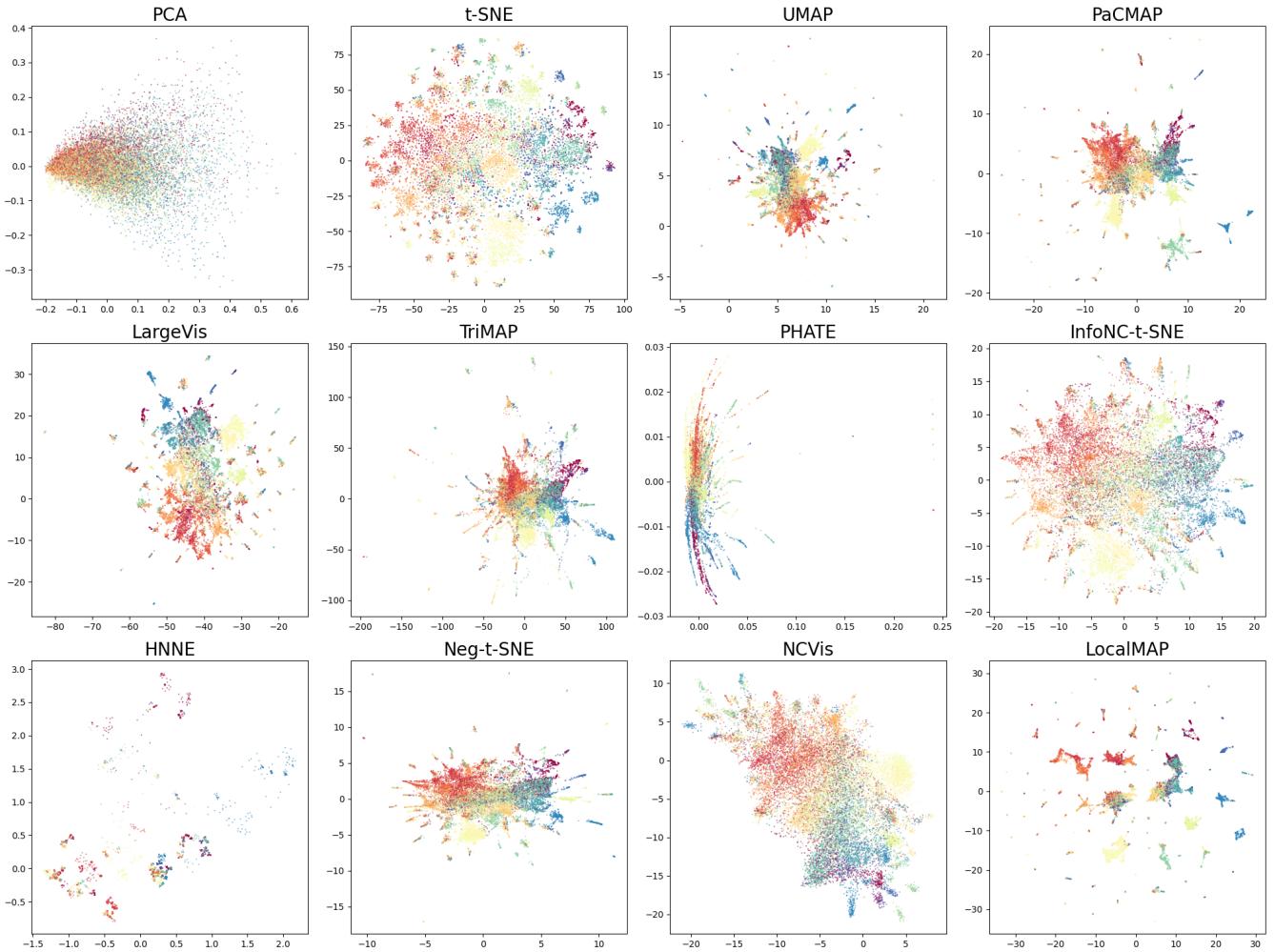


Figure 12: Different DR embeddings for 20NG (Lang 1995). Newsgroups tend to be intertwined.

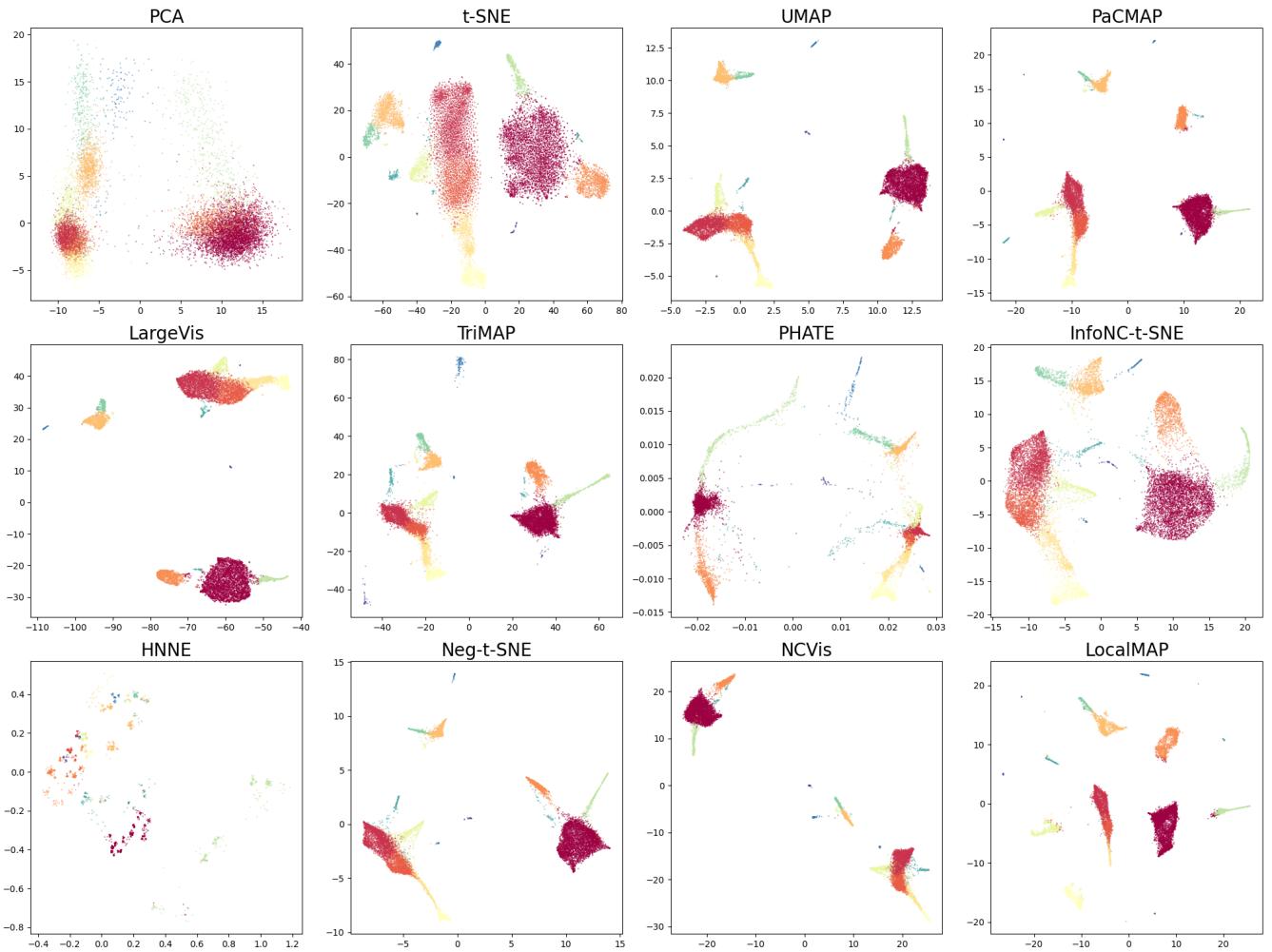


Figure 13: Different DR embeddings for Kang (Kang et al. 2018)

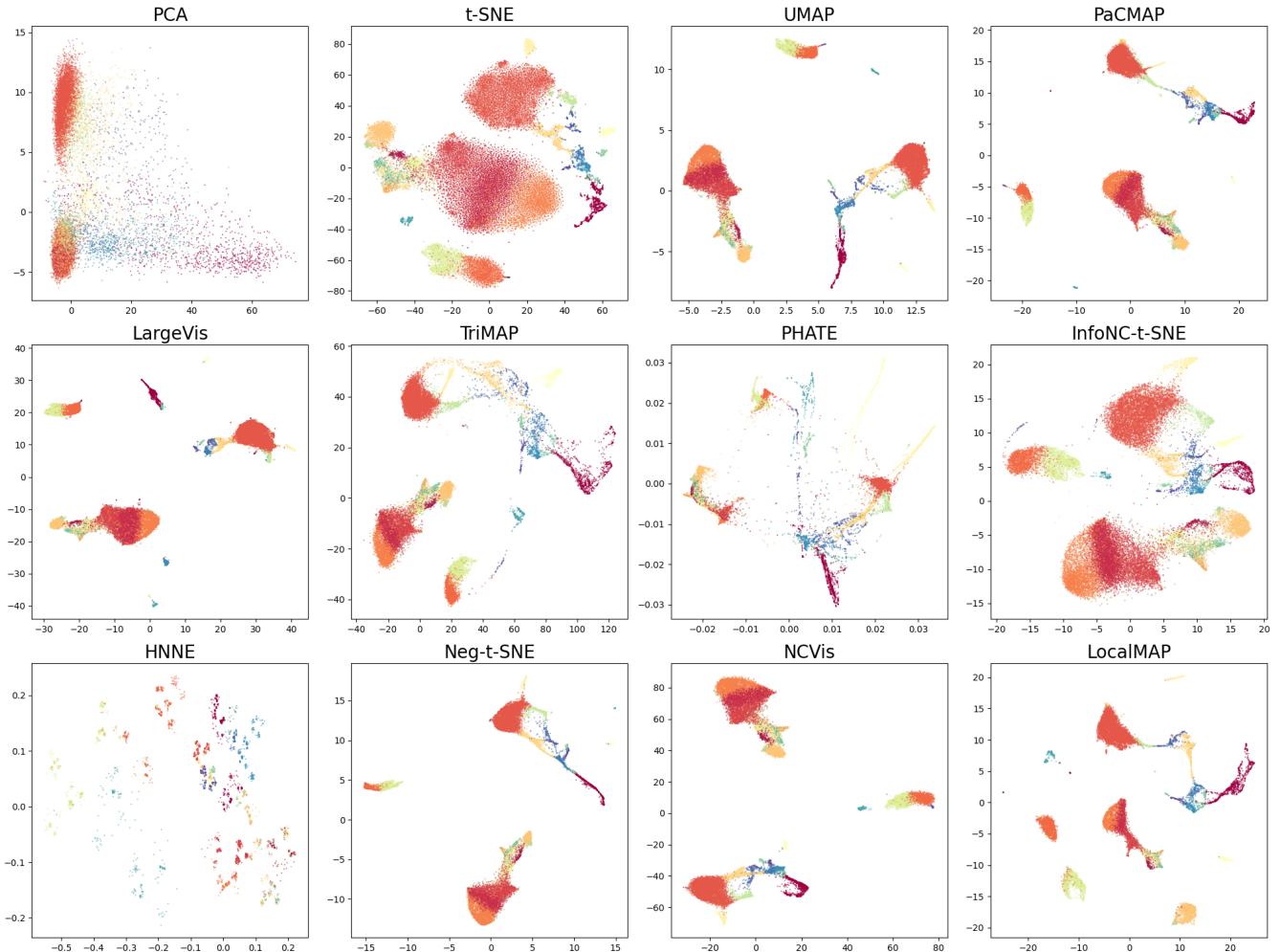


Figure 14: Different DR embeddings for Seurat (Stuart et al. 2019)

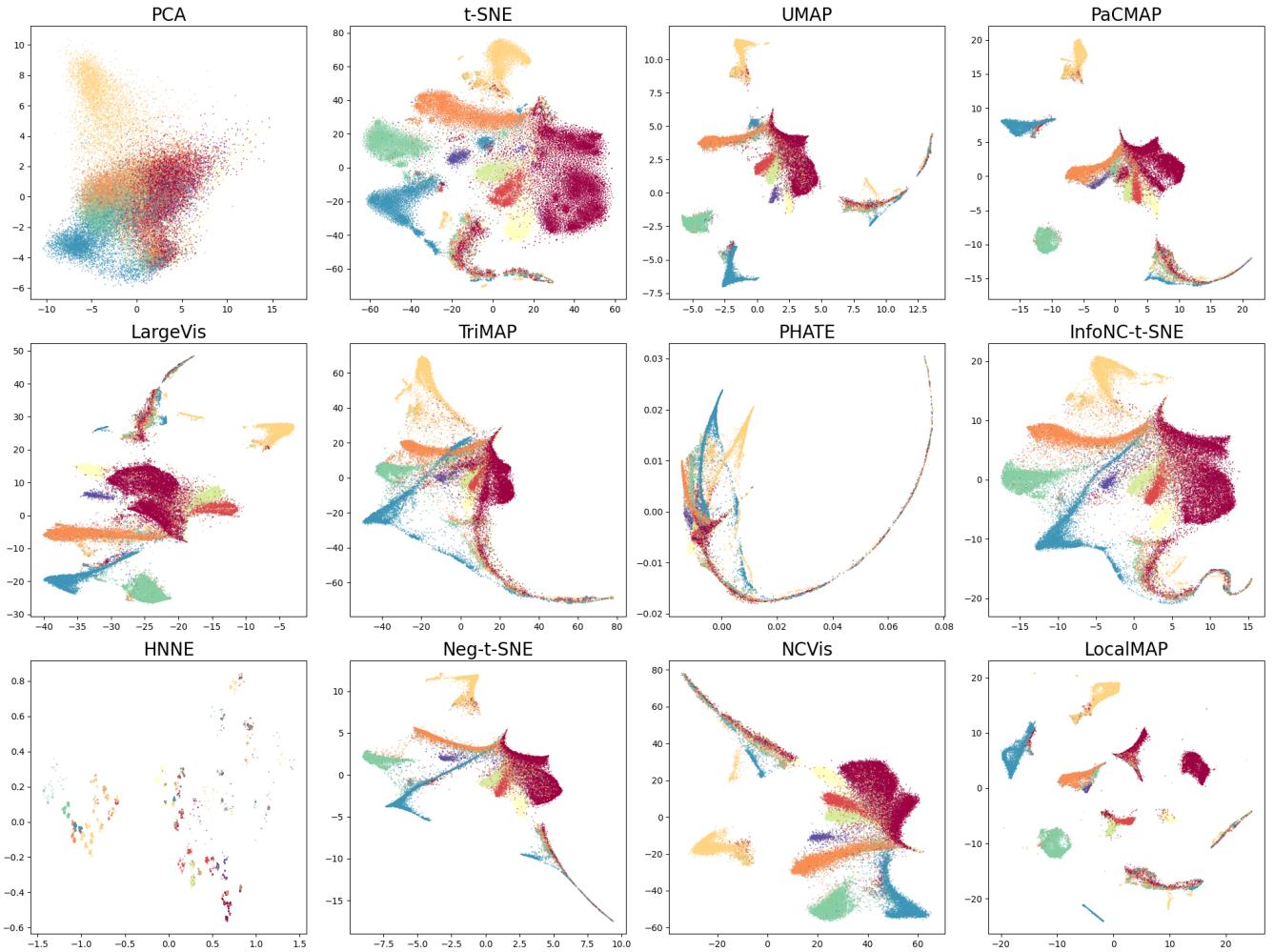


Figure 15: Different DR embeddings for Human Cortex (Zhu et al. 2023)

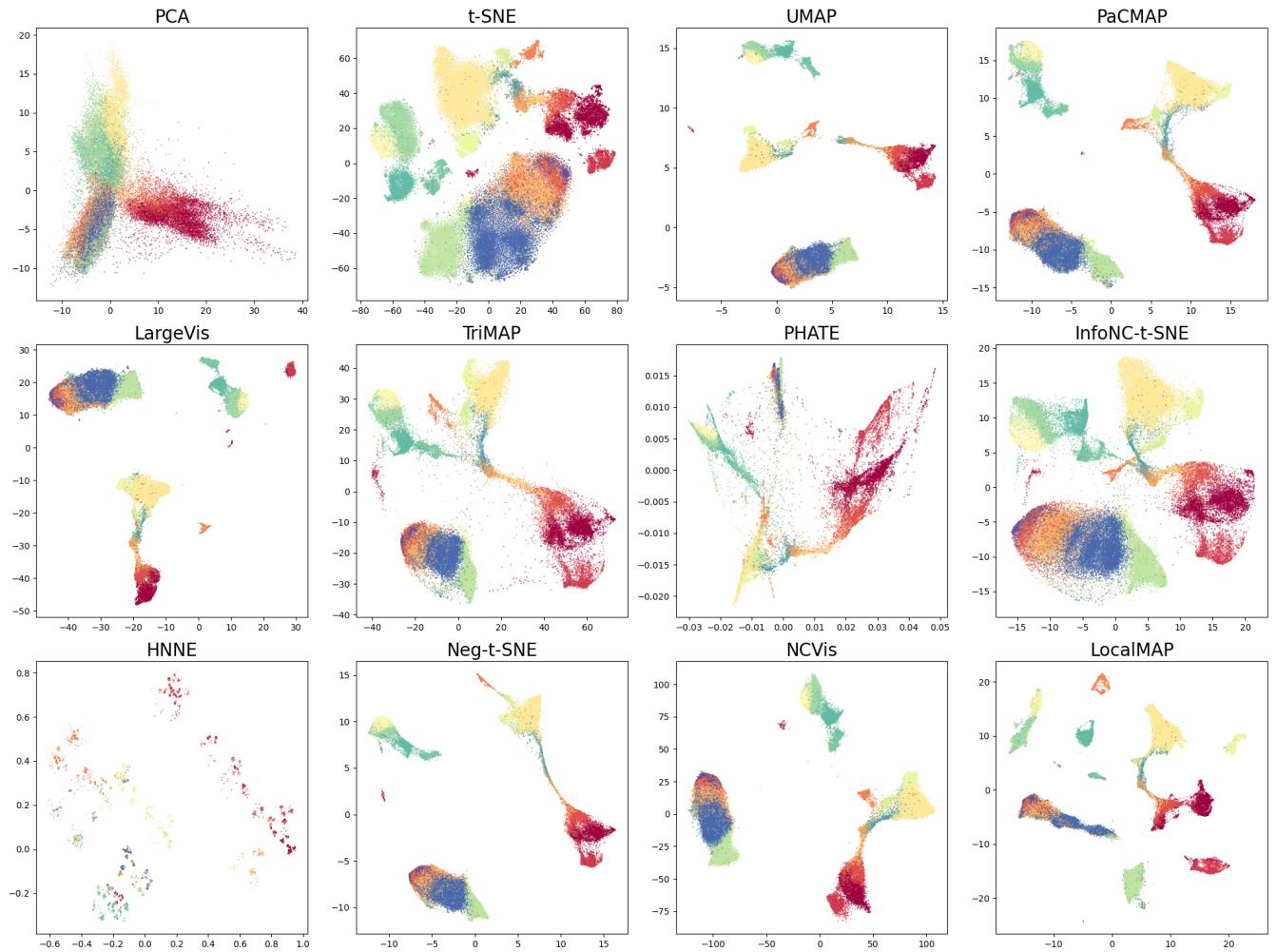


Figure 16: Different DR embeddings for CBMC (Stoeckius et al. 2017)

## I Sensitivity Check for Initialization

In this section, we will assess how stable different dimension reduction methods are when points are uniformly randomly initialized in the low-dimensional space. Figures 17 show the results. LocalMAP is able to reliably separate the 10 clusters for every run. No other method is able to do this for any run – each has multiple clusters combined that should be separated. t-SNE sometimes has severe flaws in its DR plot for MNIST in that the blue cluster is sometimes broken up; UMAP does this once for one of the red clusters. TriMAP has severe problems with local structure preservation. We have also marked those seriously problematic areas in Figure 17 with red dashed boxes.

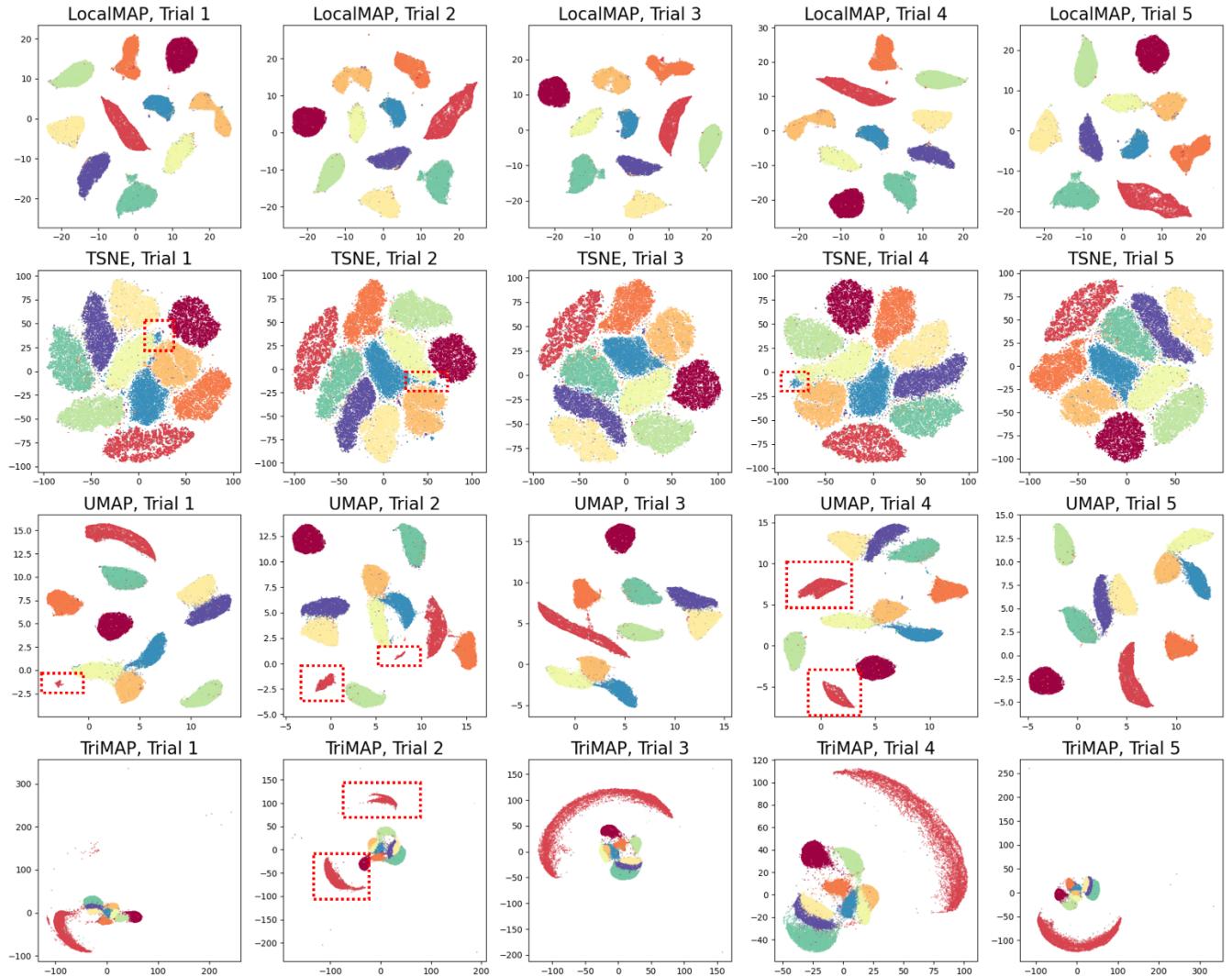


Figure 17: DR embeddings under different initializations for MNIST (LeCun, Cortes, and Burges 2010); the red dashed boxes represent the broken clusters in the embeddings.

## J Comparison with other dimension reduction methods with tuned hyperparameters

In the main paper, we have shown that LocalMAP can separate clusters better than other approaches. In this section, we compare LocalMAP with its default parameters to t-SNE, UMAP, PHATE and LargeVis with their best parameters. For tuning, we applied grid hyperparameter search, and selected the best hyperparameters among all possible combinations. For t-SNE, we tuned perplexity ([5, 10, 15, 20, 25, 30, 35, 40, 45, 50]) and learning rate ([10, 50, 100, 200, 500, 1000]) based on the suggested range from Gove et al. (2022). For UMAP, we tuned the number of nearest neighbors ([2, 5, 10, 20, 50, 100, 200]) and the min distance ([0.0, 0.1, 0.25, 0.5, 0.8, 0.99]) based on the official UMAP documentation (McInnes, Healy, and Melville 2018). For PHATE we used the number of nearest neighbors ([2, 5, 10, 15, 20]) and the decay value ([10, 15, 20, 40, 80, 160]) suggested by the original paper(Moon et al. 2019). For LargeVis, we used the range suggested by the original paper (Tang et al. 2016) and adjusted the perplexity ([10, 50, 100, 200, 500]), the number of times for neighbor propagation (prop) ([1,2,3]), and the weights assigned to negative edges ( $\gamma$ ) ([1,3,5,7,9]). The number of neighbors is chosen as three times the perplexity based on the corresponding github document. The weights assigned to negative edges in table 4 show the best parameters we found for each dataset, and table 6 shows the updated comparisons. Based on the table 6, we can easily observe that LocalMAP can still separate better than the other approaches within most of the datasets. For the COIL20 datasets, we can observe that tuned LargeVis, UMAP, and t-SNE perform slightly better than LocalMAP. However, if we look at the visualizations of these embeddings generated with the tuned hyperparameters in Figure 18, we can easily see that these methods don't provide additional separations for the clusters. Instead, they improve the silhouette score by reducing the intra-cluster distances. Moreover, if we tend to fine-tune the dimension reduction methods to achieve a better performance, it might take more time than using the default hyperparameters, which again proves that LocalMAP are less sensitive to the hyperparameters to achieve a good separation of the clusters.

Table 4: The best hyperparameters for different dimension reduction methods on different datasets with the highest silhouette score.

	<b>t-SNE (perplexity, learning rate)</b>	<b>UMAP (n_neighbors, min_dist)</b>	<b>PHATE (k,decay)</b>	<b>LargeVis (perplexity, prop,<math>\gamma</math>)</b>
<b>MNIST</b>	(50,1000)	(10,0)	(2,15)	(10,2,3)
<b>FMNIST</b>	(50,50)	(100,0)	(5,80)	(50,1,3)
<b>USPS</b>	(40,500)	(10,10)	(20,160)	(10,2,7)
<b>COIL20</b>	(10,500)	(10,0)	(5,160)	(10,1,9)
<b>20NG</b>	(40,50)	(100,0.1)	(15,160)	(10,2,1)
<b>Kang</b>	(40,500)	(10,0)	(5,20)	(10,1,1)
<b>Seurat</b>	(50,200)	(20,0)	(5,160)	(10,1,3)
<b>Human Cortex</b>	(45,1000)	(5,0)	(2,160)	(100,2,9)
<b>CBMC</b>	(45,500)	(200,0)	(2,80)	(100,1,1)

Table 5 has shown the best hyperparameters for each dimension reduction methods with the highest SVM accuracy within different datasets and table 7 has shown their corresponding SVM accuracy. Based on the performance of the dimension reduction, we can see that LocalMAP is still comparable with other optimized DR methods with respect to these scores.

Table 5: The best hyperparameters for different dimension reduction methods on different datasets with the highest SVM accuracy

	<b>t-SNE</b> (perplexity, learning rate)	<b>UMAP</b> (n_neighbors, min_dist)	<b>PHATE</b> (k,decay)	<b>LargeVis</b> (perplexity, prop, $\gamma$ )
<b>MNIST</b>	(20,200)	(20,0.1)	(5,160)	(10,3,1)
<b>FMNIST</b>	(15,1000)	(10,0.1)	(20,80)	(10,3,5)
<b>USPS</b>	(20,200)	(10,0)	(20,80)	(10,2,7)
<b>COIL20</b>	(5,10)	(5,0.1)	(2,40)	(10,2,9)
<b>20NG</b>	(20,1000)	(20,0)	(20,40)	(50,3,3)
<b>Kang</b>	(50,10)	(20,0.1)	(10,160)	(100,2,9)
<b>Seurat</b>	(50,10)	(50,0.5)	(10,80)	(50,2,5)
<b>Human Cortex</b>	(50,10)	(20,0)	(2,80)	(50,2,9)
<b>CBMC</b>	(35,50)	(20,0.5)	(2,40)	(100,1,9)

Table 6: Silhouette scores for different Algorithms. If the method is not labeled as “Optimized”, then it is using the default hyperparameters. **Bold** is best, underline is not significantly different from best. Each row is an algorithm, each column is a dataset. Red labels are the ones that have shown significant improvement comparing to LocalMAP

	<b>MNIST</b>	<b>FMNIST</b>	<b>USPS</b>	<b>COIL20</b>	<b>20NG</b>	<b>Kang</b>	<b>Seurat</b>	<b>Human Cortex</b>	<b>CBMC</b>
<b>PCA</b>	0.02±0.00	-0.03±0.00	0.10±0.00	0.01±0.00	-0.19±0.00	0.12±0.00	-0.06±0.00	-0.08±0.00	-0.11±0.00
<b>t-SNE</b>	0.35±0.00	0.12±0.00	0.42±0.00	0.41±0.00	<u>-0.11±0.00</u>	0.40±0.01	0.22±0.01	0.11±0.01	0.15±0.01
<b>UMAP</b>	0.52±0.01	0.19±0.01	0.53±0.00	0.58±0.01	<u>-0.15±0.01</u>	0.51±0.00	0.30±0.00	0.12±0.02	0.22±0.00
<b>PaCMAP</b>	0.54±0.01	<b>0.19±0.00</b>	<u>0.56±0.00</u>	0.51±0.02	<u>-0.11±0.01</u>	0.53±0.00	<u>0.31±0.00</u>	0.13±0.01	0.22±0.00
<b>LargeVis</b>	0.49±0.05	0.11±0.03	0.41±0.12	0.38±0.01	-0.13±0.01	0.44±0.01	0.25±0.01	0.10±0.02	0.17±0.00
<b>TriMAP</b>	0.41±0.00	<u>0.17±0.00</u>	0.48±0.00	0.47±0.00	-0.13±0.00	<u>0.55±0.00</u>	<b>0.32±0.00</b>	0.07±0.00	0.21±0.00
<b>PHATE</b>	0.26±0.02	0.11±0.01	0.27±0.01	0.33±0.00	-0.21±0.01	0.48±0.02	0.27±0.01	-0.09±0.01	0.06±0.01
<b>HNNE</b>	0.21±0.03	0.06±0.04	0.23±0.00	0.03±0.00	-0.34±0.03	0.39±0.06	-0.00±0.03	-0.09±0.06	0.12±0.05
<b>Neg-t-SNE</b>	0.48±0.00	<b>0.19±0.00</b>	0.48±0.00	0.44±0.01	<u>-0.11±0.00</u>	0.53±0.00	<b>0.32±0.00</b>	0.12±0.00	<b>0.24±0.00</b>
<b>NCVis</b>	0.38±0.02	<b>0.19±0.00</b>	0.44±0.00	0.53±0.00	<u>-0.15±0.00</u>	0.51±0.00	0.27±0.00	0.10±0.00	0.20±0.00
<b>InfoNC-t-SNE</b>	0.33±0.00	0.13±0.00	0.37±0.00	0.43±0.01	<u>-0.11±0.00</u>	0.46±0.00	0.26±0.00	0.10±0.00	0.21±0.00
<b>Optimized LargeVis</b>	<u>0.56±0.01</u>	<b>0.19±0.00</b>	0.55±0.03	<b>0.65±0.03</b>	-0.12±0.01	0.47±0.01	0.27±0.02	0.12±0.00	0.22±0.00
<b>Optimized PHATE</b>	0.34±0.02	0.12±0.00	0.55±0.00	0.33±0.00	-0.18±0.01	0.51±0.01	0.27±0.01	-0.01±0.00	0.10±0.02
<b>Optimized t-SNE</b>	0.39±0.00	0.14±0.00	0.43±0.01	0.51±0.00	<u>-0.11±0.00</u>	0.43±0.02	0.23±0.00	0.13±0.01	0.18±0.01
<b>Optimized UMAP</b>	<b>0.58±0.00</b>	<b>0.19±0.00</b>	<b>0.60±0.00</b>	<u>0.63±0.00</u>	-0.13±0.01	0.54±0.01	<u>0.31±0.00</u>	<b>0.14±0.00</b>	0.22±0.00
<b>LocalMAP</b>	<b>0.58±0.00</b>	<b>0.19±0.00</b>	<b>0.60±0.00</b>	0.56±0.01	<b>-0.10±0.00</b>	<b>0.60±0.00</b>	<b>0.32±0.00</b>	<b>0.14±0.00</b>	0.22±0.00

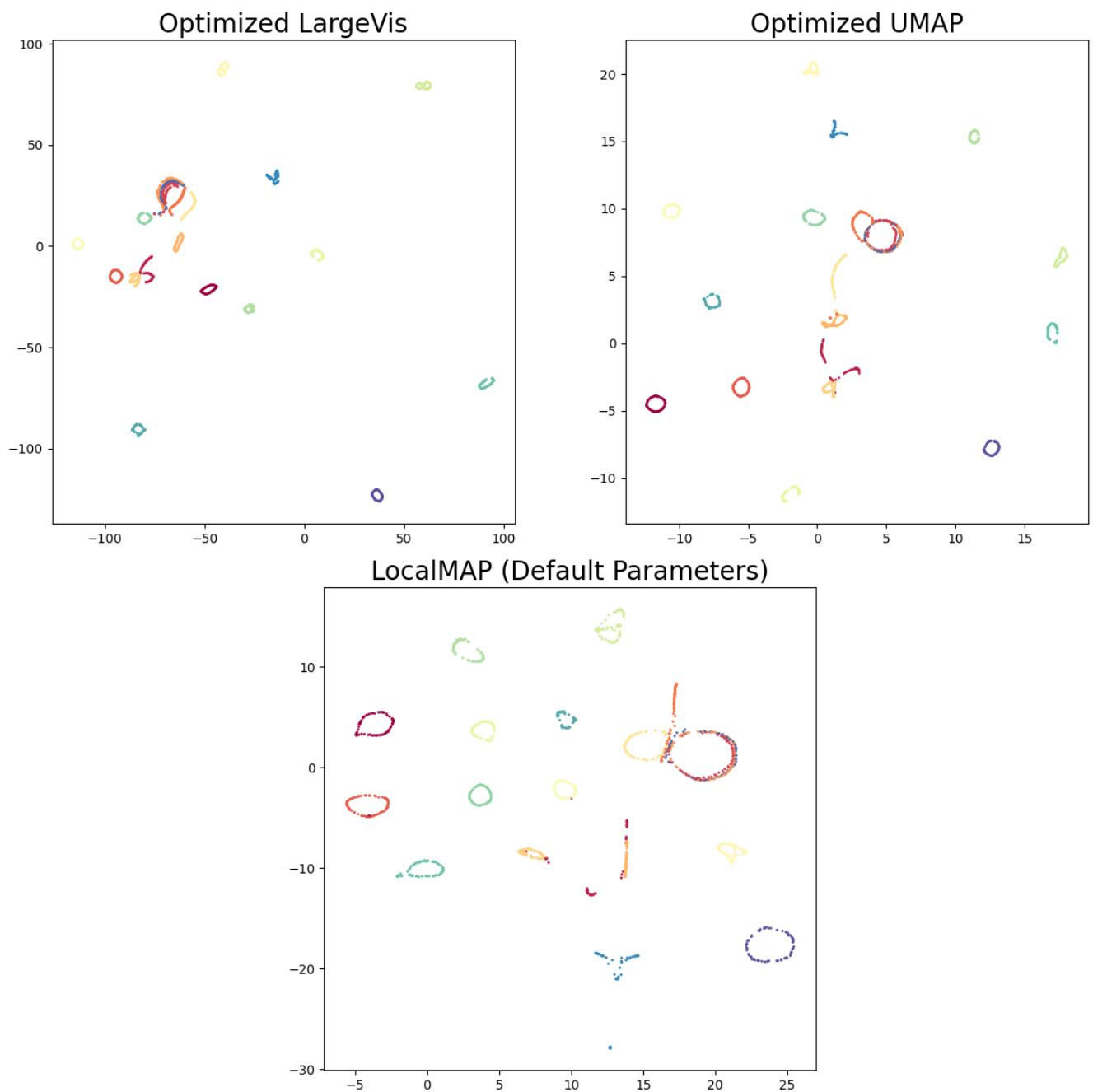


Figure 18: The visualization comparison among LocalMAP, optimized LargeVis and optimzied UMAP.

Table 7: SVM Score for Different Algorithms. If the method is not labeled as “Optimized”, then it is using the default hyperparameters. **Bold** is best, underline is not significantly different from best (with only 1% difference). Each row is an algorithm, each column is a dataset.

	MNIST	FMNIST	USPS	COIL20	20NG	Kang	Seurat	Human Cortex	CBMC
PCA	0.47±0.00	0.55±0.00	0.56±0.00	0.66±0.00	0.15±0.00	0.73±0.00	0.46±0.00	0.57±0.00	0.44±0.00
t-SNE	<b>0.97±0.00</b>	<u>0.74±0.00</u>	<b>0.96±0.00</b>	0.85±0.01	0.45±0.01	0.95±0.00	<u>0.84±0.00</u>	<b>0.82±0.00</b>	0.82±0.00
UMAP	<b>0.97±0.00</b>	<u>0.74±0.01</u>	<u>0.95±0.00</u>	0.82±0.01	0.44±0.01	0.95±0.00	0.83±0.00	<u>0.81±0.00</u>	0.82±0.00
PaCMAP	<b>0.97±0.00</b>	<u>0.74±0.00</u>	<u>0.95±0.00</u>	0.83±0.01	<u>0.46±0.01</u>	0.95±0.00	<b>0.85±0.00</b>	<u>0.81±0.00</u>	<b>0.83±0.00</b>
LargeVis	<u>0.96±0.00</u>	<u>0.74±0.01</u>	0.92±0.06	0.80±0.02	<b>0.47±0.00</b>	0.95±0.00	<u>0.84±0.00</u>	<b>0.82±0.00</b>	<u>0.82±0.00</u>
TriMAP	<u>0.96±0.00</u>	0.73±0.00	<u>0.95±0.00</u>	0.77±0.01	0.42±0.01	0.95±0.00	<u>0.84±0.00</u>	0.79±0.00	<u>0.82±0.00</u>
PHATE	0.86±0.02	0.66±0.01	0.86±0.01	0.84±0.00	0.33±0.01	0.92±0.00	0.77±0.00	0.70±0.01	0.72±0.01
HNNE	0.84±0.03	0.68±0.01	0.82±0.00	0.63±0.00	0.24±0.05	0.90±0.01	0.74±0.01	0.68±0.03	0.73±0.04
Neg-t-SNE	<u>0.96±0.00</u>	<u>0.74±0.00</u>	0.93±0.00	0.81±0.01	0.43±0.01	0.95±0.00	<u>0.84±0.00</u>	<u>0.81±0.00</u>	<u>0.82±0.00</u>
NCVis	0.94±0.01	0.73±0.00	0.92±0.00	0.79±0.00	0.36±0.01	0.94±0.00	0.83±0.00	<b>0.82±0.00</b>	<u>0.82±0.00</u>
InfoNC-t-SNE	<u>0.96±0.00</u>	<u>0.74±0.00</u>	0.93±0.00	0.82±0.01	0.42±0.00	<u>0.95±0.00</u>	<b>0.85±0.00</b>	<u>0.81±0.00</u>	<b>0.83±0.00</b>
LocalMAP	<b>0.97±0.00</b>	<u>0.75±0.00</u>	<b>0.96±0.00</b>	0.83±0.01	<u>0.46±0.01</u>	<b>0.96±0.00</b>	<u>0.84±0.00</u>	<u>0.81±0.00</u>	<u>0.82±0.00</u>
Optimized LargeVis	<b>0.97±0.00</b>	<u>0.74±0.01</u>	<b>0.96±0.00</b>	0.85±0.01	<u>0.46±0.00</u>	0.95±0.00	<u>0.84±0.00</u>	<b>0.82±0.01</b>	<u>0.82±0.00</u>
Optimized t-SNE	<b>0.97±0.00</b>	<u>0.75±0.00</u>	<b>0.96±0.00</b>	<u>0.92±0.02</u>	0.46±0.00	0.95±0.00	<u>0.84±0.01</u>	<b>0.82±0.00</b>	<u>0.82±0.01</u>
Optimized UMAP	<b>0.97±0.00</b>	<u>0.75±0.00</u>	<b>0.96±0.00</b>	0.89±0.00	0.44±0.01	0.95±0.00	<u>0.84±0.00</u>	<b>0.82±0.00</b>	<u>0.82±0.00</u>
Optimized PHATE	0.89±0.01	0.68±0.02	0.86±0.00	<b>0.93±0.01</b>	0.37±0.00	0.92±0.01	0.77±0.00	0.71±0.01	0.72±0.00
LocalMAP	<b>0.97±0.00</b>	<u>0.75±0.00</u>	<b>0.96±0.00</b>	0.83±0.01	<u>0.46±0.01</u>	<b>0.96±0.00</b>	0.84±0.00	<u>0.81±0.00</u>	0.82±0.00

## K Scalability of LocalMAP under Large Datasets

In this section, we have added two large single-cell datasets with more than 1 million cells within each dataset to show the scalability of our model. The data description of the extended dataset within our model has already been shown in Table 8. The biological data sets are processed according to the same method mentioned in Section 8, and the detailed embeddings using PaCMAp and LocalMAP are shown in Figure 19, which proves that LocalMAP shows good separation even in large-scale settings.

Table 8: Data Description with Large Scale Dataset over 1 million samples

Dataset	# of samples	# of dimensions
PBMC 1M(Perez et al. 2022)	1,263,676	1,000
AIDA(Kock et al. 2024)	1,058,909	1,000

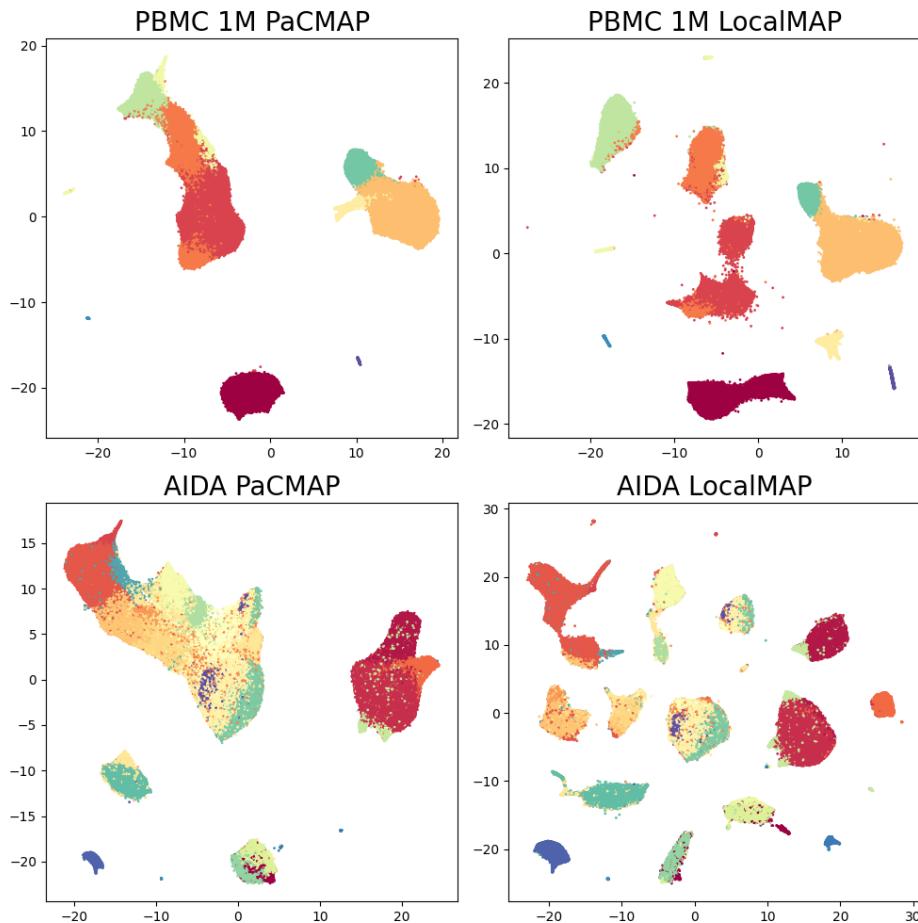


Figure 19: The performance of PaCMAp and LocalMAP under large scale settings