

# THE SHAPE OF ATTRACTION IN UMAP: EXPLORING THE EMBEDDING FORCES IN DIMENSIONALITY REDUCTION

**Anonymous authors**

Paper under double-blind review

## ABSTRACT

Uniform manifold approximation and projection (UMAP) is among the most popular neighbor embedding methods. The method samples pairs of point indices according to similarities in the high-dimensional space, and applies attractive and repulsive forces to their coordinates in the low-dimensional embedding. In this paper, we analyze the forces to reveal their effects on cluster formations and visualization, and compare UMAP to its contemporaries. Repulsion emphasizes differences, controlling cluster boundaries and inter-cluster distance. Attraction is more subtle, as attractive tension between points can manifest simultaneously as attraction and repulsion in the lower-dimensional mapping. This explains the need for learning rate annealing and motivates the different treatments between attractive and repulsive terms. Moreover, by modifying attraction, we improve the consistency of cluster formation under random initialization. Overall, our analysis makes UMAP and similar embedding methods more interpretable, more robust, and more accurate.

## 1 INTRODUCTION

Modern applications routinely generate high-dimensional data. Dimensionality reduction (DR) techniques have emerged as tools for exploratory analysis of such data by visualizing the underlying structure. The most popular methods,  $t$ -distributed stochastic neighbor embedding (Maaten & Hinton, 2008) and uniform manifold approximation and projection (UMAP) (McInnes et al., 2018) are grounded in the attraction-repulsion dynamics that bring similar data points closer while pushing dissimilar ones apart. As unsupervised algorithms, these do not rely on labeled data; instead, they identify and preserve the intrinsic structure of high-dimensional data by leveraging local (attractive) and global (repulsive) relationships (forces). This makes these algorithms particularly well-suited for tasks such as clustering (Becht et al., 2019), exploratory data analysis (Fleischer & Islam, 2020), anomaly detection in semiconductor manufacturing (Fan et al., 2021), visual search (González-Márquez et al., 2024), time series analysis (Altin & Cakir, 2024), studying representation convergence (Huh et al., 2024), and outlier image detection (Islam & Fleischer, 2024), where visualizing hidden patterns in unlabeled data is critical and meaningful. By learning the embeddings in a data-driven, label-free manner, DR exemplifies the power of unsupervised methods to distill complex data into easily interpretable forms.

Building upon the attraction-repulsion principle, newer methods have emerged (Amid & Warmuth, 2019; Agrawal et al., 2021; Wang et al., 2021; Narayan et al., 2021; Yang et al., 2022; Wang et al., 2025; Kury et al., 2025), each designed to emphasize specific aspects of the data. Despite their relevance in diverse applications, these methods often rely on heuristic practices that may fail to give meaningful interpretations. Moreover, DR introduces distortions that are unavoidable (Chari & Pachter, 2023). Thus, it is imperative to have a deeper understanding of the algorithms so that practitioners can provide better interpretations of the embeddings, avoid spurious structures, and optimize performance. In practice, these algorithms achieve compact clusters using a variety of techniques, including specific initialization, learning rate schedule, and kernel function tuning. However, the underlying dynamics of the attractive and repulsive forces, responsible for cluster formation, have not been thoroughly investigated. Furthermore, the essential tunable parameters are concealed within abstract functional forms, making it harder to explain the algorithms.

In this paper, we decompose the forces into their constituent parts and extract the functional shapes of attraction and repulsion. We find that the necessity of learning rate annealing, the challenge of providing consistent output under random initialization, and the origin of cluster formation rely on attraction. Repulsive forces primarily govern inter-cluster distances. Our specific contributions are:

1. We formulate attraction and repulsion shapes from the attractive and repulsive forces, establish the conditions for contraction and expansion of distance, and provide a fresh perspective for these algorithms (Section 4).
2. We show that the attraction shape of UMAP causes the counterintuitive concept of both contraction and expansion of distance. Comparing attraction shapes of different algorithms, we discuss how attraction influences the learning rate annealing scheme (Section 5).
3. We modify the attraction shape to improve the consistency of embedding under random initialization. This indicates the encoding of a unique structure (Section 5.1).
4. Analyzing repulsion shapes, we provide a deeper understanding of cluster formation and regulating inter-cluster distance (Section 5.2).
5. We compare attraction and repulsion shapes of UMAP, Parametric UMAP, and NEG-*t*-SNE, unveil the similarities and distinctions among them, and characterize the stability of the algorithms (Section 6).

We center the main text on UMAP, now a de facto standard in many fields, and bring in other algorithms where relevant. For further details on different algorithms, see the appendix.

## 2 RELATED WORKS

The origin of modern iterative graph-based neighbor embedding algorithms can be traced to stochastic neighbor embedding (SNE) (Hinton & Roweis, 2002) and its extension using the *t*-distribution (*t*-SNE) (Maaten & Hinton, 2008). Both methods use a dense graph in which each point in a dataset has a pairwise relation with all the others, regardless of whether they are similar to each other or not. Moreover, the weights of the graphs are normalized to give a notion of probability distribution. Other concurrent methods, including locally linear embedding (Roweis & Saul, 2000) and Laplacian Eigenmaps (LE) (Belkin & Niyogi, 2002), used a *k*-nearest neighbor (*k*-NN) graph of pairwise interaction. Known as spectral methods, these algorithms rely on Eigenvalue decomposition. Subsequent work by Tang et al. (Tang et al., 2016) incorporated the *k*-NN graph in the iterative approach and removed normalization in the lower dimension. This approach was further extended by McInnes et al. (2018) in UMAP, where the normalization step was removed altogether (both in high and low dimension) and the embedding was obtained using pairwise interactions alone. The optimization steps use an explicit attractive force to preserve the local neighborhood and a repulsive force to keep dissimilar points apart. Building on these foundations, methods such as PaCMAP (Wang et al., 2021) and NEG-*t*-SNE (Damrich et al., 2023) have been proposed. For a recent survey of methods, see de Bodt et al. (2025).

There has been considerable progress in understanding and explaining the relationship among these algorithms. An early analysis of SNE found that if the data is well-clustered in the original space, then they are well-clustered in the embedding space (Shaham & Steinerberger, 2017). A similar analysis for *t*-SNE by Linderman & Steinerberger (2019) showed that the number of clusters in the embedding space is a lower bound on the number of clusters in the original space. This was followed up in further characterization (Arora et al., 2018; Cai & Ma, 2022; Linderman & Steinerberger, 2022). Since *t*-SNE and UMAP originate from the same underlying framework (but with drastically different visualizations), a major undertaking in the literature has been to find the connection between them (Böhm et al., 2022; Damrich et al., 2023; Draganov et al., 2023). Böhm et al. (2022) theorized that methods like Laplacian Eigenmap, ForceAtlas2 (Jacomy et al., 2014), UMAP, and *t*-SNE are all samples from the same underlying spectrum. Indeed, LE and *t*-SNE’s are connected by the early exaggeration phase (Cai & Ma, 2022). The connection between UMAP and *t*-SNE can be related through contrastive estimation (Damrich et al., 2023). Around the same time, Hu et al. (2023) independently discovered the relation of contrastive learning and SNE. Recent approaches offer a probabilistic perspective (Ravuri et al., 2023; Ravuri & Lawrence, 2024), employ kernel techniques (Draganov & Dohn, 2023), and utilize information geometry (Kolpakov & Rocke, 2024) to explain dimensionality reduction.

### 3 UNIFORM MANIFOLD APPROXIMATION AND PROJECTION

UMAP constructs a high-dimensional graph of the dataset  $X = \{x_i \in \mathbb{R}^n | i = 1, \dots, N\}$  by using a pairwise relation:  $p_{i,j} = f_h(d(x_i, x_j))$ , typically,  $\in [0, 1]$ , **where,  $f_h$  is the high dimensional affinity function and  $d(\cdot, \cdot)$  is a distance metric** (for details, see Appendix H).

The graph of the low-dimensional representation  $Y = \{y_i \in \mathbb{R}^d | i = 1, \dots, N\}$  is given by a differentiable function

$$q_{ij} = \frac{1}{1 + a(\|y_i - y_j\|_2^2)^b}, \quad (1)$$

where the parameters  $a$  and  $b$  determine the density of the mapping and are chosen by fitting  $q_{ij}$  to

$$\Psi(\|y_i - y_j\|_2) = \begin{cases} 1 & \text{if } \|y_i - y_j\|_2 < m_d \\ \exp(-(\|y_i - y_j\|_2 - m_d)) & \text{otherwise} \end{cases}, \quad (2)$$

where  $m_d$  regulates the distance between the two nearest low-dimensional points.

UMAP aims to minimize the following cross-entropy loss function:

$$\mathcal{L} = \sum_{i,j} (-p_{ij} \log(q_{ij}) - (1 - p_{ij}) \log(1 - q_{ij})). \quad (3)$$

The first term provides an attractive force and the second term provides a repulsive force. Instead of optimizing every point in each iteration, UMAP takes the negative sampling approach (Mikolov et al., 2013; Tang et al., 2016). For each edge with  $p_{ij} > 0$ , named a positive edge, several edges are sampled randomly, named negative edges. The attractive force is applied on the positive edge:

$$y_i^{t+1} = y_i^t + \lambda \nabla_{y_i^t} \log(q_{ij}), \quad (4)$$

$$y_j^{t+1} = y_j^t + \lambda \nabla_{y_j^t} \log(q_{ij}), \quad (5)$$

and the repulsive force is applied on the negative edges:

$$y_i^{t+1} = y_i^t + \lambda \nabla_{y_i^t} \log(1 - q_{ij}), \quad (6)$$

where  $\lambda (> 0)$  is the learning rate and  $t$  is the step number. Note that  $y_j$  is not updated for negative edges. For a detailed analysis of UMAP's loss, see Damrich & Hamprecht (2021).

### 4 ATTRACTION AND REPULSION SHAPES

The action of the updates (4-6) can be simplified by decomposing the gradients  $\nabla_{y_i^t} \log(q_{ij})$  ( $\nabla_{y_i^t} \log(1 - q_{ij})$ ) into a scalar coefficient dependent on the distance  $\|y_i - y_j\|_2$  acting on the vector  $(y_i - y_j)$ . We call this scalar coefficient the attraction (repulsion) shape. While we use UMAP as a specific example, this formalism applies to any method that relies on attraction and repulsion.

By writing  $\nabla_{y_i^t} \log(q_{ij}) = f_a(\zeta^t)(y_i^t - y_j^t)$ , where  $\zeta = \|y_i - y_j\|_2$ , we can update the equations of a positive edge as

$$y_i^{t+1} = y_i^t + \lambda f_a(\zeta^t)(y_i^t - y_j^t), \quad (7)$$

$$y_j^{t+1} = y_j^t - \lambda f_a(\zeta^t)(y_i^t - y_j^t). \quad (8)$$

Here,  $f_a : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\leq 0}$  is the attraction shape, and we use the fact that for Euclidean metric,  $\nabla_{y_i^t} \log(q_{ij}) = -\nabla_{y_j^t} \log(q_{ij})$ . Similarly, we reformulate the update equation of a negative edge as

$$y_i^{t+1} = y_i^t + \lambda f_r(\zeta^t)(y_i^t - y_j^t), \quad (9)$$

$$y_j^{t+1} = y_j^t, \quad (10)$$

where  $f_r : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$  is the repulsion shape.

Such decomposition has appeared previously, e.g., in (McInnes et al., 2018; Agrawal et al., 2021; Draganov & Dohn, 2023), but their formulations and utilization vary. The original UMAP paper (McInnes et al., 2018) used it as a computational trick for fast processing, while Draganov

& Dohn (2023) used it for comparing different algorithms from the kernel perspective. In both cases, the primary focus was computing the derivative. Here, we treat the decompositions as independent functions that can take various forms. A similar approach was used earlier by Agrawal et al. (2021), but they expressed the decomposition in terms of a scalar coefficient and a unit vector  $((y_i - y_j)/\|y_i - y_j\|_2)$  to emphasize the magnitude of the forces. The discussion below shows that our shape decomposition is more illuminating than the magnitude alone.

#### 4.1 CONDITIONS FOR ATTRACTION AND REPULSION

In this section, we establish the conditions of attraction and repulsion from the update equations (7)-(10). The following proposition characterizes the contraction of distance between the pair  $y_i^t$  and  $y_j^t$  of a positive edge:

**Proposition 4.1.** *The update Eqs. (7) and (8) provide a contraction of distance ( $\|y_i^{t+1} - y_j^{t+1}\| < \|y_i^t - y_j^t\|$ ) if  $-1 < \lambda f_a < 0$ .*

Here,  $\lambda f_a$  works as the effective attraction shape. In most cases, however,  $f_a$  alone is enough to draw meaningful conclusions about the embeddings. For a negative edge, the following proposition characterizes the expansion of the distance between the pair  $y_i^t$  and  $y_j^t$ :

**Proposition 4.2.** *The update Eqs. (9) and (10) provide an expansion of distance ( $\|y_i^{t+1} - y_j^{t+1}\| > \|y_i^t - y_j^t\|$ ) if  $f_r > 0$ .*

Note that the inclusion of a symmetric term  $-\lambda f_r(\zeta)(y_i^t - y_j^t)$  in Eq. (10) does not alter the conclusion presented in this proposition. Additionally, these conditions give a per-iteration certificate for guaranteed contraction/expansion, and one should not confuse them with the learning rate tuning/decay mechanism, as we can design the shapes in such a way that the contraction and expansion do not rely on learning rate decay.

Figure 1 (a) shows the effect of different values of  $f_a < 0$  and  $f_r > 0$  on two points. The latter shape is straightforward, as any positive value increases the distance. The former is much more subtle, as  $f_a$  encodes both attractive and repulsive dynamics. For  $f_a \in (-1.0, 0)$ , the distance decreases, with a sign flip at the value  $f_a = -0.5$  of maximum attraction (coincident points). Any value lower than  $-1.0$  causes the distance to increase. Although these forces act locally, they collectively shape the global structure.

## 5 ANALYSIS OF UMAP IN TERMS OF ATTRACTION AND REPULSION SHAPE

Using the gradient decomposition and the distance form (1), the attraction and repulsion shapes are given by

$$f_a^U(\zeta) = -\frac{2ab\zeta^{2(b-1)}}{1 + a\zeta^{2b}} \quad (11)$$

and

$$f_r^U(\zeta) = \frac{2b}{\zeta^{2b}(1 + a\zeta^{2b})}, \quad (12)$$

respectively. This section focuses on the default shapes of UMAP, controlled primarily by the parameters  $a$  and  $b$ , and discusses the insights learned by perturbing them. The discussion below is generally valid for  $b \leq 1$ , where the attraction shape is strictly increasing and thus invertible (For derivation, see Appendix A).

Figure 1 (b) shows the default attraction shape of UMAP ( $a = 1.58, b = 0.89$ ). It becomes unbounded (approaches  $-\infty$ ) as  $\zeta \rightarrow 0$ . As predicted by Proposition 4.1, the transition from contraction to expansion occurs when  $\lambda f_a^U$  crosses  $-1$  as  $\zeta$  approaches 0. Since the attraction shape is invertible, we can identify the distance at this transition,  $\zeta_{-1} = (\lambda f_a^U)^{-1}(-1)$ , as the minimum distance for contraction due to attractive updates. Effectively,  $\zeta > \zeta_{-1}$  causes contraction, and  $0 < \zeta < \zeta_{-1}$  causes expansion, contradicting the intuition that attractive updates consistently bring points closer together.

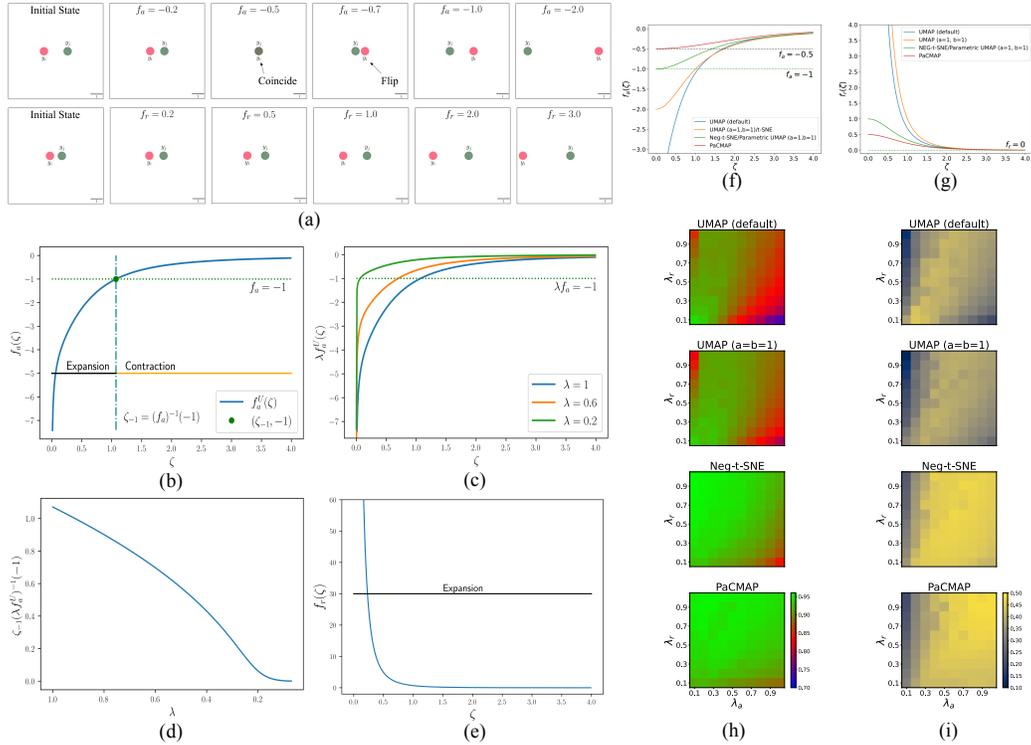


Figure 1: Attraction and repulsion shapes in UMAP. (a) Effect of different values of  $f_a$  (top) and  $f_r$  (bottom) on a pair. (b) Attraction shape of UMAP. (c) Effective attraction shape ( $\lambda f_a$ ) for various learning rates  $\lambda$ . (d) Minimum distance for contraction ( $\zeta_{-1}$ ) as  $\lambda$  decreases. (e) Repulsion shape of UMAP. (f) Attraction and (g) repulsion shapes of various embedding methods. (h) Trustworthiness and (i) Silhouette Score of various methods as the constant learning rate is varied for attraction ( $\lambda_a$ ) and repulsion ( $\lambda_r$ ) independently. Default UMAP parameters:  $a = 1.58$  and  $b = 0.89$ .

If  $\zeta_{-1}$  is high, neighboring points oscillate between contraction and expansion, and the clusters appear fuzzy. For the sharpest boundaries, then, the goal of optimization can be recast as one of achieving the limit  $\zeta_{-1} \rightarrow 0$ . **The default shape oscillates around  $\zeta = 1.07$ , which is large compared to the expectation.** As a result, UMAP’s learning rate schedule requires annealing to zero (Figs. 1 (c,d)). On the other hand, the repulsion shape (Eq. 12) is always positive and satisfies Proposition 4.2.  $f_r^U$  approaches 0 as  $\zeta \rightarrow \infty$  and approaches  $\infty$  as  $\zeta \rightarrow 0$  (Fig. 1 (e)).

Attraction shapes for different algorithms show that only UMAP and t-SNE deal with the issue of having  $f_a < -1$  (and consequently  $\zeta_{-1} > 0$  at  $\lambda = 1$ , Fig. 1(f)). t-SNE solves it by weighting the updates with corresponding  $p_{ij}$  values, while UMAP relies on learning rate annealing. Methods like Neg-t-SNE and PaCMAP have  $f_a$  naturally within  $[-1, 0]$  (and thus, satisfies Proposition 4.1 for  $\lambda \in [0, 1]$  with  $\zeta_{-1} = 0$ ). Furthermore, PaCMAP’s weighted  $f_a$  ( $< 0.5$  always) even prevents any flips during attraction. On the other hand, the repulsion shapes for different algorithms show that only UMAP deals with large values for small distances (Fig. 1(g); the shape is unbounded, but during optimization, the values are often clipped).

We show the effect of these attraction/repulsion choices for MNIST embedding by applying separate constant learning for attraction ( $\lambda_a$ ) and repulsion ( $\lambda_r$ ), and varying their values (Figs. 1(h,i)). For UMAP, the lower value of  $\lambda_a$  ( $< 0.5$  and consequently  $\zeta_{-1} \simeq 0$ ) is preferred for better embedding (trustworthiness (T) (Venna & Kaski, 2001), a measure of structure preservation, in Fig. 1(h)) and clustering (silhouette scores (SIL) (Rousseeuw, 1987), a measure of cluster separation, in Fig. 1(i)), whereas the whole range of parameters ( $\in [0, 1]$ ) could be used. For NEG-t-SNE and PaCMAP, for which  $\zeta_{-1} = 0$ , the whole range of parameters is effective. **This confirms that UMAP relies on making  $\zeta_{-1}$  close to 0 for satisfactory embedding and clustering, which it achieves in practice by learning rate annealing, whereas  $\zeta_{-1} = 0$  is the default in newer algorithms.**

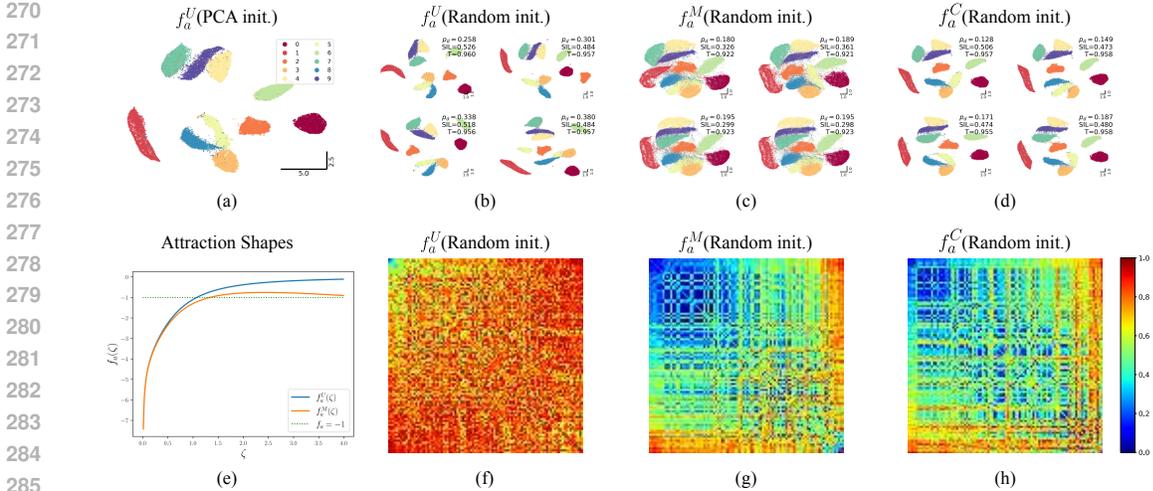


Figure 2: Effect of random initialization on different attraction shapes for the MNIST dataset. (a) Mapping using PCA. (b-d) Four mappings with the lowest Procrustes distance ( $p_d$ ) from the embedding in (a) for (b) UMAP, (c) modified, and (d) composite attraction shapes. (e) Default UMAP and modified attraction shapes. (f-h) Procrustes matrices from 100 runs of (f) UMAP ( $0.78 \pm 0.13$ ), (g) modified ( $0.49 \pm 0.21$ ), and (h) composite attraction ( $0.50 \pm 0.20$ ) shapes. The diagonal ( $i, i$ ) entries of the Procrustes matrix are sorted by Procrustes distance ( $p_d$ ) from (a), and the off-diagonal, ( $i, j$ ), entries correspond to  $p_d$  between  $i^{th}$  and  $j^{th}$  mapping. The matrices and (mean  $p_d \pm \text{std}$ ) values show that UMAP’s embeddings are not self-similar, while the modified and composite attraction shapes encourage initialization-invariant structure.

Future sections and appendices provide additional details. In Section 6 we focus on the case of  $a = 1$  and  $b = 1$  and compare UMAP to NEG-t-SNE. Appendix B provides additional discussion regarding constant learning rate in UMAP. Appendix G provides details of other dimensionality reduction algorithms (TriMAP, PaCMAP, LocalMAP, t-SNE, SNE and multidimensional scaling) from the perspective of attraction and repulsion shape. Appendix I gives details of the embeddings used in this section as well as extends results for two other datasets.

### 5.1 IMPROVING UMAP’S CONSISTENCY UNDER RANDOM INITIALIZATION

The consistency of UMAP embeddings depends on proper initialization (Kobak & Linderman, 2021; Wang et al., 2021). Typically, principal component analysis (PCA) of the data or spectral decomposition of the high-dimensional graph initializes the embedding, producing consistent mappings despite various sources of stochasticity. If randomly initialized, clusters often fail to form or form in a random orientation each time the algorithm executes. If the initial distance between two points (nearest neighbors in high dimension) is large, the attractive forces become too low to bring them closer. Known as near-sightedness (Wang et al., 2021), this phenomenon is evident in the attraction shape, where  $|f_a^U|$  diminishes towards zero as the distance increases (since,  $|f_a^U| = o(1/\zeta)$ , and thus,  $\lim_{\zeta \rightarrow \infty} |f_a^U(\zeta)|\zeta = 0$ ).

One can induce “far-sightedness” in the mapping by increasing attraction for large distances, facilitating faraway neighbors to come closer. To test this hypothesis, we modify the attraction shape of UMAP to increase the attractive force:

$$f_a^M = f_a^U - \beta\zeta, \tag{13}$$

where  $\beta$  is a parameter that regulates the strength of the added term (we used  $\beta = 0.2$ , Fig. 2 (e)). This addition in the attraction shape translates to adding a regularizer in the attractive term of the loss function (i.e.,  $\mathcal{L}^M = \mathcal{L} + \sum_{i,j} \beta/3 \|y_i - y_j\|_2^2$ ). In addition to attracting pairs at faraway distances, this technique enables intermixing of points that help convergence under random initialization (akin to early exaggeration in t-SNE). In (13), we chose the simplest linear correction; other functions, such as  $\log \zeta$  or  $\zeta^p$  ( $p \in \mathbb{R}_{\geq 0}$ ), may also be suitable.

Table 1: Effect of random initialization on different datasets using different attraction shapes and comparison to PCA initialization. The metrics (mean±std) are reported based on 100 runs of each. Datasets: MNIST (LeCun et al., 2010), FMNIST (Xiao et al., 2017), Transcriptomes- (Macosko et al., 2015), (Shekhar et al., 2016), and 20NewsGroup (20NG) (Mitchell, 1997)

Dataset	Shape (initialization)	Embedding Quality			Run to Run Consistency	
		Trustworthiness	Silhouette Score	Spearman	Spearman	Procrustes Distance
MNIST	Default (PCA, standard)	0.957 ± 0.001	0.51 ± 0.01	0.31 ± 0.01	0.95 ± 0.06	0.13 ± 0.05
	Default (Rand)	0.958 ± 0.001	0.47 ± 0.04	0.24 ± 0.06	0.44 ± 0.12	0.78 ± 0.13
	Modified (Rand)	0.923 ± 0.003	0.33 ± 0.03	0.33 ± 0.03	0.71 ± 0.12	0.49 ± 0.21
	Composite (Rand)	0.956 ± 0.001	0.48 ± 0.02	0.29 ± 0.04	0.70 ± 0.11	0.50 ± 0.21
FMNIST	Default (PCA, standard)	0.975 ± 0.001	0.18 ± 0.00	0.60 ± 0.00	0.99 ± 0.00	0.02 ± 0.00
	Default (Rand)	0.976 ± 0.001	0.11 ± 0.05	0.42 ± 0.07	0.54 ± 0.13	0.72 ± 0.15
	Modified (Rand)	0.959 ± 0.004	0.16 ± 0.03	0.57 ± 0.04	0.82 ± 0.11	0.38 ± 0.19
	Composite (Rand)	0.975 ± 0.001	0.18 ± 0.03	0.53 ± 0.05	0.78 ± 0.10	0.43 ± 0.18
Macosko	Default (PCA, standard)	0.950 ± 0.001	0.42 ± 0.04	0.77 ± 0.01	0.95 ± 0.02	0.16 ± 0.07
	Default (Rand)	0.950 ± 0.001	0.23 ± 0.06	0.75 ± 0.02	0.76 ± 0.03	0.91 ± 0.06
	Modified (Rand)	0.935 ± 0.001	0.15 ± 0.05	0.71 ± 0.02	0.82 ± 0.05	0.61 ± 0.13
	Composite (Rand)	0.949 ± 0.001	0.28 ± 0.05	0.73 ± 0.02	0.83 ± 0.04	0.64 ± 0.15
Shekhar	Default (PCA, standard)	0.974 ± 0.000	0.52 ± 0.01	0.51 ± 0.01	0.94 ± 0.03	0.07 ± 0.03
	Default (Rand)	0.974 ± 0.000	0.41 ± 0.08	0.48 ± 0.03	0.52 ± 0.09	0.70 ± 0.16
	Modified (Rand)	0.965 ± 0.001	0.58 ± 0.03	0.40 ± 0.02	0.81 ± 0.06	0.48 ± 0.19
	Composite (Rand)	0.973 ± 0.000	0.51 ± 0.03	0.48 ± 0.01	0.73 ± 0.08	0.51 ± 0.19
20NG	Default (PCA, standard)	0.819 ± 0.001	-0.17 ± 0.00	0.58 ± 0.00	0.96 ± 0.01	0.05 ± 0.01
	Default (Rand)	0.818 ± 0.002	-0.18 ± 0.01	0.57 ± 0.01	0.92 ± 0.03	0.24 ± 0.13
	Modified (Rand)	0.777 ± 0.001	-0.16 ± 0.00	0.59 ± 0.00	0.95 ± 0.04	0.20 ± 0.13
	Composite (Rand)	0.817 ± 0.002	-0.18 ± 0.01	0.58 ± 0.01	0.94 ± 0.04	0.19 ± 0.15

We also consider a composite attraction shape:

$$f_a^C = \begin{cases} f_a^M, & \text{epoch} \leq 100 \\ f_a^U, & \text{otherwise} \end{cases} \quad (14)$$

The composite shape attempts to remove any distortions introduced by  $f_a^M$  by reverting to the original UMAP. Below, we discuss the effects these modified and composite attraction shapes have on DR from random initialization.

We first created a PCA-initialized embedding of the MNIST dataset Figure 2 (a). Then, we produced embeddings using random initialization (Gaussian) for each shape and repeated the experiment 100 times. To quantify the results, we use Procrustes analysis (Gower, 1975) that aligns two point clouds under scaling, translation, rotation, and reflection (for details, see Appendix D.1). Here, we align the randomly initialized embeddings to that of the PCA-initialized one and characterize their separation using the Procrustes distance ( $p_d$ ). Figure 2 (b) shows four embeddings with the lowest  $p_d$ . While the cluster shapes are consistent, their placements are not. Outputs from the modified and composite attraction shapes (Figs. 2 (c) and (d), respectively) show improved consistency of cluster placements.

To quantify the placements further, we consider the Procrustes matrix: the diagonal of the matrix is sorted by  $p_d$  from the PCA-initialized mapping, and the off-diagonal values are  $p_d$  between two randomly initialized mappings (for details, see Appendix D.1). This quantification is analogous to the similarity matrix (Foote, 1999). The embeddings due to the default UMAP attraction shape are not similar to each other (Fig. 2 (f)), but the modified (Fig. 2 (g)) and composite (Fig. 2 (h)) shapes show strong similarity to each other.

**Table 1 shows embedding quality and run-to-run variance of five different datasets when randomly initialized and compare it to the corresponding PCA initialized version (standard and the most consistent one). Overall, we can observe that the embedding quality is comparable among different shapes. Overall, modified and composite shapes improve consistency in terms of Spearman’s rank correlation and Procrustes distance. This indicates that UMAP, regardless of the initialization, aims to encode a unique structure (in our experiments, PCA initialized embedding is an attractor). However, attaining that structure in the low dimension may fall short due to small attraction at longer distances.**

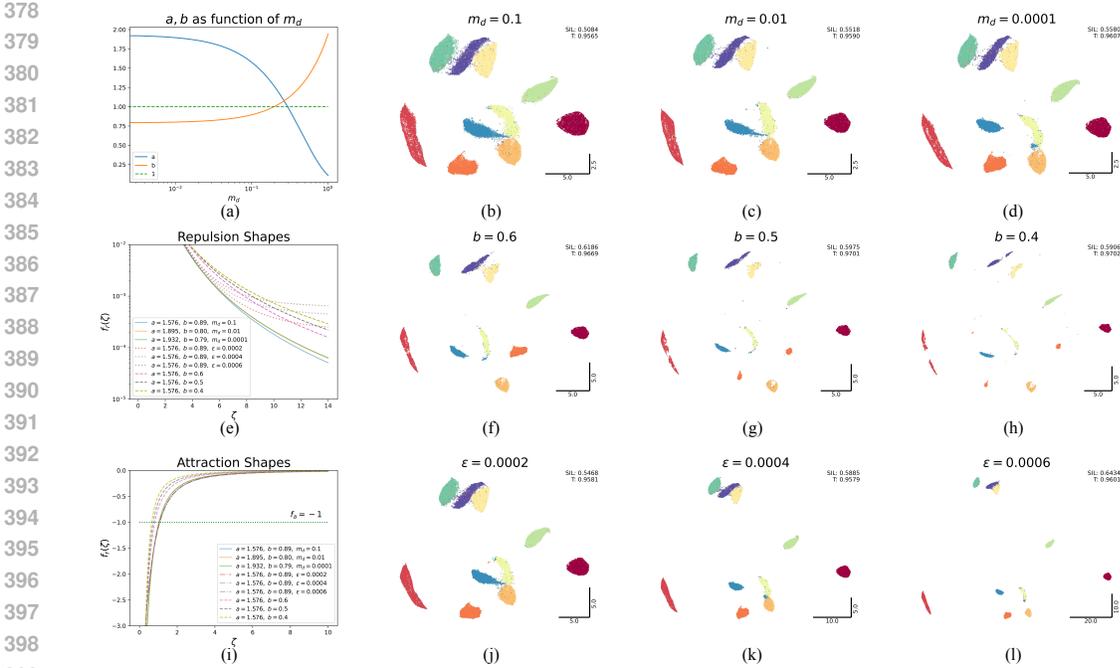


Figure 3: Control of inter-cluster distances on the MNIST dataset. (a) Computing  $a, b$  by varying the low-dimensional distance  $m_d$  restricts exploration. (b-d) UMAP output by setting  $m_d$  to 0.1, 0.01, and 0.0001, respectively, shows little improvement in compactness of clusters. (e) Repulsion shapes for different parameters. (f-h) Increasing repulsion by explicitly varying  $b$  results in more compact clusters and forms new ones that were absent otherwise. (i) Attraction shapes by varying parameters. (j-l) Increasing repulsion by adding a small positive value ( $\epsilon$ ) to the repulsion shape increases inter-cluster distance.

## 5.2 CLUSTER FORMATION AND COMPACTNESS

The primary controllable parameter influencing cluster formation in UMAP is the minimum distance parameter  $m_d$  (through Eq. 2). However, varying  $m_d$  restricts the exploration of different values of  $a$  and  $b$  (Fig. 3 (a)). Thus, reducing  $m_d$  often results in embeddings that do not provide additional benefit (Figs. 3 (b-d)). The key factor is the limited influence of varying  $m_d$  on the repulsion shape (Fig. 3 (e)). Alternatively, we can explicitly vary the values of  $a$  and  $b$ . Decreasing  $a$  increases repulsion, but it decreases attraction at a faster rate (causing a worse case of near-sightedness). On the other hand, decreasing  $b$  gives a better control (Figs. 3 (e,i)). Figures 3 (f-h) show increasing inter-cluster distance and breaking up of previous clusters by varying  $b$  to 0.6, 0.5, and 0.4, respectively. This breaking up occurs due to the increasingly heavy-tailed nature of the kernel as  $b$  decreases (heavy-tailed kernels result in smaller and distinct clusters (Van der Maaten & Hinton, 2008; Kobak et al., 2019); see Kobak et al. (2019) for how varying  $b$  in UMAP modulates the tail). Although this approach separates all ten MNIST labels into distinct clusters, the relative contributions of attraction and repulsion are hard to disentangle; changing  $b$  amplifies repulsion more than changing  $m_d$ , but it also reshapes the attraction profile.

To quantify the effect of repulsion independently, we can keep attraction fixed and vary the other. To achieve this, we modify the repulsion shape by adding a small positive value ( $\epsilon$ ):

$$f_r^M = f_r^U + \epsilon, \tag{15}$$

while keeping the values of  $a$  and  $b$  constant, which effectively adds a regularizer in the repulsive term of the loss function (i.e.  $\mathcal{L}^M = \mathcal{L} - \sum_{i,j} \epsilon/2 \|y_i - y_j\|_2^2$ ). Using this, we can obtain stronger repulsion than previously (Fig. 3(e)). Figs. 3 (j-l) show that as  $\epsilon$  increases, the inter-cluster distances also increase. However, the clustering properties show similarity to those obtained by varying  $m_d$ , and we get a loose separation of all the labels as  $\epsilon$  increases. Overall, the parameter  $\epsilon$  keeps the

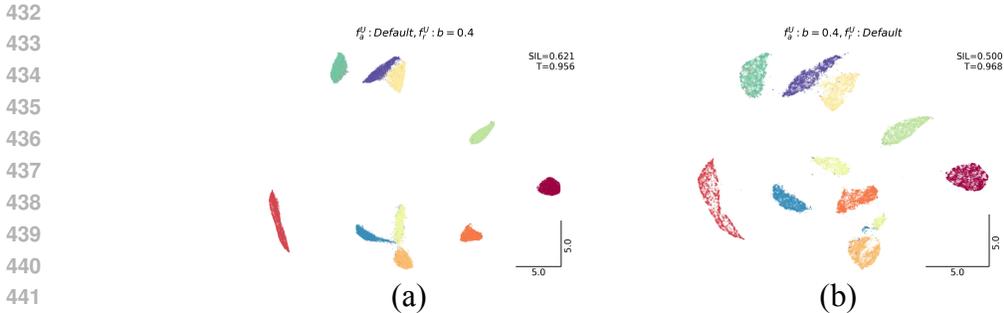


Figure 4: Embedding of the MNIST dataset with (a) default attraction shape but repulsion shape with  $b = 0.4$  and (b) default repulsion shape but attraction shape with  $b = 0.4$ . The former shows the same clusters of default UMAP with increased compactness. The latter develops new structures within clusters and forms new clusters.

attraction shape unaffected, and varying  $m_d$  effectively traces similar attraction shapes (Fig. 3 (i)), suggesting cluster formation is governed predominantly by attraction.

To resolve the mystery of cluster formation, we change either the attraction shape or the repulsion shape individually while leaving the other at the default by simply setting  $b$  to 0.4 (Fig. 4). The default attraction is unable to show new structures or clusters in the embedding, but the increased repulsion ( $b = 0.4$ ) gives smaller clusters than the original UMAP (Fig. 4 (a)). On the other hand, when the attraction increases by setting  $b = 0.4$  with the default repulsion, the embedding shows additional structures within each cluster (Fig. 4 (b)). Some of the older clusters even separate into smaller ones. This shows that attraction causes cluster formation, while repulsion makes the clusters more compact (depending on the repulsion shape, it does this either by making smaller clusters or increasing inter-cluster distance). **By controlling attraction and repulsion behavior explicitly, we can create embeddings with various levels of granularity.**

**As an aside, one of the claims of LocalMAP is to separate the ten labels of MNIST into individual clusters. Here, we have achieved the same result in UMAP by simply manipulating the attraction and the repulsion shapes. In Appendix G.3, we explore how LocalMAP embeddings exploit this interplay of attraction and repulsion to separate the clusters and relate it to the experiments of this section.**

## 6 COMPARISON TO NEG- $t$ -SNE

Setting  $a$  and  $b$  to 1 is common for various dimensionality reduction algorithms. This particular setting makes the gradients in many algorithms stable (or bounded) and, in turn, makes optimization easier. Recently, Damrich et al. (2023) explored this for UMAP and proposed Neg- $t$ -SNE as a solution, which, in addition to having a stable gradient, provides a more compact clustering even for a constant learning rate of 1. On the other hand, Parametric UMAP (Sainburg et al., 2021) initially used the UMAP loss formulation, but later it<sup>1</sup> adopted a numerically stable modified cross-entropy loss function (Shi et al., 2023) with a logsigmoid kernel. Analysis of both reveals that the approaches arrive at the same formulation from different starting points, which leads to the following observation:

**Proposition 6.1** (Damrich et al. (2023)). *Neg- $t$ -SNE is Parametric UMAP with  $a = 1$  and  $b = 1$ .*

With  $a = 1$  and  $b = 1$ , the attraction and repulsion shapes of UMAP are given by  $f_a^U = -2/(1 + \zeta^2)$  and  $f_r^U = 2/(\zeta^2(1 + \zeta^2))$ , respectively. The attraction shape becomes bounded within  $[-2, 0]$ , with  $f_a^U(0) = -2$  (Fig. 1 (f)), while the repulsion shape remains essentially unchanged (i.e., unbounded as  $\zeta \rightarrow 0$ , Fig. 1 (g)). Since  $f_a^U < -1$  as  $\zeta \rightarrow 0$ , according to Proposition 4.1 and the discussion provided in Section 5, this unity case still requires learning rate annealing.

Damrich et al. (2023) compared UMAP’s negative sampling loss function from the perspective of contrastive embedding (CE) and concluded that the effective kernel of UMAP is  $1/\zeta^2$ . Under

<sup>1</sup><https://github.com/lmcinnes/umap/pull/856>, merged on April 26, 2022

the CE framework, the authors introduced NEG- $t$ -SNE by changing the kernel to  $1/(1 + \zeta^2)$ . Reverting to UMAP formalism, this results in the low-dimensional affinity function  $q_{ij}^N = 1/(2 + \zeta^2)$ . Consequently, the attraction and repulsion shapes are  $f_a^N = -2/(2 + \zeta^2)$ , and  $f_r^N = 2/((1 + \zeta^2)(2 + \zeta^2))$ , respectively (Figs. 1 (f,g)). The attraction shape is bounded within  $[-1, 0]$  and satisfies Proposition 4.1. Any  $\lambda \in [0, 1]$  would cause contraction and avoid oscillation of expansion and contraction. Thus, NEG- $t$ -SNE is less sensitive to learning rate annealing, and the clusters appear less fuzzy even for constant  $\lambda = 1.0$ . Moreover, the repulsion shape of NEG- $t$ -SNE is also bounded within  $[0, 1]$  and does not approach infinity as  $\zeta \rightarrow 0$ , which causes fewer points to leave the clusters when sampled randomly. While Damrich et al. (2023) used only the bounded repulsive forces of NEG- $t$ -SNE and unbounded ones of UMAP to explain this disparity, our analysis, in Fig. 1, shows that the attraction shape is equally responsible for stability and compactness of the clusters. **When  $a$  and  $b$  vary in NEG- $t$ -SNE, it faces the same numerical challenges of UMAP.**

For additional discussion on NEG- $t$ -SNE with illustration, see Appendix F. Together with the discussion in Fig. 4, this suggests that attraction and repulsion shapes from different well-behaved methods (i.e., that guarantee Proposition 4.1) can be mixed to achieve desired embedding properties, for instance, using the attraction shape of UMAP with the repulsion shape of NEG- $t$ -SNE. We show a few examples by experimenting on UMAP, NEG- $t$ -SNE, and PaCMAP in Appendix J. **We empirically show that  $f_a \in [-1, 0]$  achieves better clustering, while  $f_a \in [-0.5, 0]$  (prevents flips) obtains better local structure.**

## 7 DISCUSSION AND CONCLUSION

In this work, we studied the relationship of attractive and repulsive forces to cluster formation. While it is known that attractive forces bring similar points closer together and repulsive forces push dissimilar points further away, the exact mechanisms of such forces have not been well studied. Here, we have analyzed the dynamics underlying cluster formation. While we focused on UMAP and NEG- $t$ -SNE in the main text, the results are general to dimensionality reduction (some other algorithms are discussed in the Appendix G).

We treated the attraction and repulsion coefficients as functional mappings and analyzed them in the context of their respective shapes. Characterizing these shapes (Propositions 4.1 and 4.2) revealed a counterintuitive result: the attractive forces of UMAP gave repulsion (expansion of distance) for shorter distances (Fig. 1 (b)), i.e., instead of bringing neighboring points closer together, it pushes them away. We conclude that UMAP’s learning rate schedule alleviates this phenomenon.

We also studied the initialization of the low-dimensional embedding. Similar points starting further apart experience low attraction and never contract well enough. Our formalism provides a way to influence attraction at larger distances by adding additional terms to the attraction shape. This modification resulted in outputs that are more consistent under random initialization. This gives us confidence that UMAP encodes a unique structure, with numerical tricks merely helping to achieve it faster. Analyzing the repulsion shape revealed that higher repulsion causes a larger inter-cluster distance without changing the cluster characteristics.

**Overall, the key takeaways for the practitioners are:**

1. **Attraction and repulsion shapes characterize pair-wise dimensionality reduction methods and provide formal guidance for contraction/expansion of points. Various methods can be put on the same graph, and their properties can be described (Fig. 1(f,h)).**
2. **For effective contraction, attraction shapes that follow Proposition 4.1 are desirable. If it doesn’t, learning rate tuning is required (UMAP uses annealing).**
3. **Near-sightedness in the attraction force hinders consistent output. Tweaking its shape to include long-range attraction can mitigate this issue.**
4. **To increase inter-cluster distance modifying repulsion shape is enough (Eq. 15). To break up clusters, one may modify just the attraction shape.**

These insights into attraction–repulsion dynamics offer new tools for optimizing dimensionality reduction algorithms. Beyond this, the close connection between dimensionality reduction and contrastive learning (Damrich et al., 2023; Hu et al., 2023) suggests that our approach can also enhance representation learning. Taken together, our work aims to make embeddings and their interpretations more principled, consistent, and reliable, and guide future research.

540 SOFTWARE AND DATA

541  
542 All the data used in this paper are publicly available. The MNIST dataset is  
543 available at <https://yann.lecun.com/exdb/mnist/>. Fashion-MNIST is available at  
544 <https://github.com/zalandoresearch/fashion-mnist>. Single-cell transcriptomes data is available at  
545 <https://github.com/biolab/tsne-embedding>. Additional details are provided in the Implementation  
546 Details section in Appendix K. Representative codes for reproducing the results are attached as  
547 supplementary material for review.

548  
549 LLM USAGE DISCLOSURE

550  
551 The first author drafted the manuscript and used LLM tools (Grammarly and ChatGPT) for stylistic  
552 and grammatical refinement in select sections. All authors subsequently edited and approved the final  
553 text. No AI tools were used for code generation, data analysis, or research assistance.

554  
555 REFERENCES

- 556  
557 Akshay Agrawal, Alnur Ali, Stephen Boyd, et al. Minimum-distortion embedding. *Foundations and*  
558 *Trends® in Machine Learning*, 14(3):211–378, 2021.
- 559  
560 Mahsun Altin and Altan Cakir. Exploring the influence of dimensionality reduction on anomaly  
561 detection performance in multivariate time series. *IEEE Access*, 2024.
- 562  
563 Ehsan Amid and Manfred K Warmuth. TriMap: Large-scale dimensionality reduction using triplets.  
564 *arXiv preprint arXiv:1910.00204*, 2019.
- 565  
566 Sanjeev Arora, Wei Hu, and Pravesh K Kothari. An analysis of the t-sne algorithm for data visualiza-  
567 tion. In *Conference On Learning Theory*, pp. 1455–1462. PMLR, 2018.
- 568  
569 Etienne Becht, Leland McInnes, John Healy, Charles-Antoine Dutertre, Immanuel WH Kwok,  
570 Lai Guan Ng, Florent Ginhoux, and Evan W Newell. Dimensionality reduction for visualizing  
571 single-cell data using umap. *Nature biotechnology*, 37(1):38–44, 2019.
- 572  
573 Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps and spectral techniques for embedding and  
574 clustering. In *Advances in neural information processing systems*, pp. 585–591, 2002.
- 575  
576 Jan Niklas Böhm, Philipp Berens, and Dmitry Kobak. Attraction-repulsion spectrum in neighbor  
577 embeddings. *The Journal of Machine Learning Research*, 23(1):4118–4149, 2022.
- 578  
579 Ingwer Borg and Patrick JF Groenen. *Modern multidimensional scaling: Theory and applications*.  
580 Springer Science & Business Media, 2007.
- 581  
582 T Tony Cai and Rong Ma. Theoretical foundations of t-sne for visualizing high-dimensional clustered  
583 data. *Journal of Machine Learning Research*, 23(301):1–54, 2022.
- 584  
585 Tara Chari and Lior Pachter. The specious art of single-cell genomics. *PLOS Computational Biology*,  
586 19(8):e1011288, 2023.
- 587  
588 Sebastian Damrich and Fred A Hamprecht. On UMAP’s true loss function. *Advances in Neural*  
589 *Information Processing Systems*, 34:5798–5809, 2021.
- 590  
591 Sebastian Damrich, Niklas Böhm, Fred A Hamprecht, and Dmitry Kobak. From t-SNE to UMAP  
592 with contrastive learning. In *The Eleventh International Conference on Learning Representations*,  
593 2023.
- 594  
595 Cyril de Bodt, Alex Diaz-Papkovich, Michael Bleher, Kerstin Bunte, Corinna Coupette, Sebastian  
596 Damrich, Enrique Fita Sanmartin, Fred A Hamprecht, Emőke-Ágnes Horvát, Dhruv Kohli, et al.  
597 Low-dimensional embeddings of high-dimensional data. *arXiv preprint arXiv:2508.15929*, 2025.
- 598  
599 Andrew Draganov and Simon Dohn. Relating tsne and umap to classical dimensionality reduction.  
600 *arXiv e-prints*, pp. arXiv–2306, 2023.

- 594 Andrew Draganov, Jakob Jørgensen, Katrine Scheel, Davide Mottin, Ira Assent, Tyrus Berry, and  
595 Cigdem Aslay. ActUp: Analyzing and consolidating tSNE and UMAP. In *Proceedings of the*  
596 *Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23*, pp. 3651–3658,  
597 8 2023.
- 598  
599 Shu-Kai S Fan, Du-Ming Tsai, Chih-Hung Jen, Chia-Yu Hsu, Fei He, and Li-Ting Juan. Data  
600 visualization of anomaly detection in semiconductor processing tools. *IEEE Transactions on*  
601 *Semiconductor Manufacturing*, 35(2):186–197, 2021.
- 602 Jason Fleischer and Mohammad Tariqul Islam. Late breaking abstract-identifying and phenotyping  
603 COVID-19 patients using machine learning on chest x-rays. *European Respiratory Journal*, 2020.
- 604  
605 Jonathan Foote. Visualizing music and audio using self-similarity. In *Proceedings of the seventh*  
606 *ACM international conference on Multimedia (Part 1)*, pp. 77–80, 1999.
- 607 Rita González-Márquez, Luca Schmidt, Benjamin M Schmidt, Philipp Berens, and Dmitry Kobak.  
608 The landscape of biomedical research. *Patterns*, 2024.
- 609  
610 John C Gower. Generalized procrustes analysis. *Psychometrika*, 40:33–51, 1975.
- 611  
612 Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann,  
613 Jianfeng Gao, and Hoifung Poon. Domain-specific language model pretraining for biomedical  
614 natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3(1):  
615 1–23, 2021.
- 616 Geoffrey Hinton and Sam T Roweis. Stochastic neighbor embedding. In *Advances in Neural*  
617 *Information Processing Systems*, volume 15, pp. 833–840, 2002.
- 618  
619 Tianyang Hu, Zhili Liu, Fengwei Zhou, Wenjia Wang, and Weiran Huang. Your contrastive learning  
620 is secretly doing stochastic neighbor embedding. In *The Eleventh International Conference on*  
621 *Learning Representations*, 2023.
- 622  
623 Minyoung Huh, Brian Cheung, Tongzhou Wang, and Phillip Isola. Position: The platonic representa-  
624 tion hypothesis. In *Proceedings of the 41st International Conference on Machine Learning*, pp.  
625 20617–20642, 2024.
- 626  
627 Mohammad Tariqul Islam and Jason W Fleischer. Manifold-aligned neighbor embedding. In *ICLR*  
628 *2022 Workshop on Geometrical and Topological Representation Learning*, 2022.
- 629  
630 Mohammad Tariqul Islam and Jason W Fleischer. Outlier detection in large radiological datasets  
631 using umap. In *International Workshop on Topology-and Graph-Informed Imaging Informatics*,  
pp. 111–121. Springer, 2024.
- 632  
633 Robert A Jacobs. Increased rates of convergence through learning rate adaptation. *Neural networks*,  
1(4):295–307, 1988.
- 634  
635 Mathieu Jacomy, Tommaso Venturini, Sebastien Heymann, and Mathieu Bastian. Forceatlas2, a  
636 continuous graph layout algorithm for handy network visualization designed for the gephi software.  
637 *PloS one*, 9(6):e98679, 2014.
- 638  
639 Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International*  
640 *Conference on Learning Representations*, 2015.
- 641  
642 Dmitry Kobak and George C Linderman. Initialization is critical for preserving global data structure  
in both t-sne and umap. *Nature biotechnology*, 39(2):156–157, 2021.
- 643  
644 Dmitry Kobak, George Linderman, Stefan Steinerberger, Yuval Kluger, and Philipp Berens. Heavy-  
645 tailed kernels reveal a finer cluster structure in t-sne visualisations. In *Joint European Conference*  
646 *on Machine Learning and Knowledge Discovery in Databases*, pp. 124–139. Springer, 2019.
- 647  
Alexander Kolpakov and Aidan Rocke. The information geometry of umap. *Le Matematiche*, 79(1):  
151–164, 2024.

- 648 Nikita Kotlov, Kirill Shaposhnikov, Cagdas Tazearslan, Madison Chasse, Artur Baisangurov, Svet-  
649 lana Podsvirova, Dawn Fernandez, Mary Abdou, Leznath Kaneunyenyne, Kelley Morgan, et al.  
650 Procrustes is a machine-learning approach that removes cross-platform batch effects from clinical  
651 rna sequencing data. *Communications Biology*, 7(1):392, 2024.
- 652 Joseph B Kruskal. Nonmetric multidimensional scaling: a numerical method. *Psychometrika*, 29(2):  
653 115–129, 1964.
- 654 Noël Kury, Dmitry Kobak, and Sebastian Damrich. DREAMS: Preserving both local and global  
655 structure in dimensionality reduction. *arXiv preprint arXiv:2508.13747*, 2025.
- 656 Siu Kwan Lam, Antoine Pitrou, and Stanley Seibert. Numba: A llvm-based python jit compiler. In  
657 *Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC*, pp. 1–6, 2015.
- 658 Pierre Lambert, Cyril De Bodt, Michel Verleysen, and John A Lee. Squadmds: a lean stochastic  
659 quartet mds improving global structure preservation in neighbor embedding like t-sne and umap.  
660 *Neurocomputing*, 503:17–27, 2022.
- 661 Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database. *ATT Labs [Online]*.  
662 Available: <http://yann.lecun.com/exdb/mnist>, 2, 2010.
- 663 George C Linderman and Stefan Steinerberger. Clustering with t-SNE, provably. *SIAM Journal on*  
664 *Mathematics of Data Science*, 1(2):313–332, 2019.
- 665 George C Linderman and Stefan Steinerberger. Dimensionality reduction via dynamical systems:  
666 The case of t-sne. *SIAM Review*, 64(1):153–178, 2022.
- 667 Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine*  
668 *learning research*, 9(Nov):2579–2605, 2008.
- 669 Evan Z Macosko, Anindita Basu, Rahul Satija, James Nemesh, Karthik Shekhar, Melissa Goldman,  
670 Itay Tirosh, Allison R Bialas, Nolan Kamitaki, Emily M Martersteck, et al. Highly parallel genome-  
671 wide expression profiling of individual cells using nanoliter droplets. *Cell*, 161(5):1202–1214,  
672 2015.
- 673 Leland McInnes, John Healy, and James Melville. UMAP: Uniform manifold approximation and  
674 projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.
- 675 Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations  
676 of words and phrases and their compositionality. In *Advances in neural information processing*  
677 *systems*, pp. 3111–3119, 2013.
- 678 Tom Mitchell. Twenty Newsgroups. UCI Machine Learning Repository, 1997. DOI:  
679 <https://doi.org/10.24432/C5C323>.
- 680 Ashwin Narayan, Bonnie Berger, and Hyunghoon Cho. Assessing single-cell transcriptomic variabil-  
681 ity through density-preserving data visualization. *Nature Biotechnology*, 39(6):765–774, 2021.
- 682 F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Pretten-  
683 hofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and  
684 E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*,  
685 12:2825–2830, 2011.
- 686 Pavlin G. Poličar, Martin Stražar, and Blaž Zupan. opentsne: A modular python library for t-sne  
687 dimensionality reduction and embedding. *Journal of Statistical Software*, 109(3):1–30, 2024.
- 688 Aditya Ravuri and Neil D Lawrence. Towards one model for classical dimensionality reduction: A  
689 probabilistic perspective on umap and t-sne. *arXiv preprint arXiv:2405.17412*, 2024.
- 690 Aditya Ravuri, Francisco Vargas, Vidhi Lalchand, and Neil D Lawrence. Dimensionality reduction  
691 as probabilistic inference. In *ICML 2023 Workshop on Structured Probabilistic Inference* {\&}  
692 *Generative Modeling*, 2023.
- 693 Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis.  
694 *Journal of computational and applied mathematics*, 20:53–65, 1987.

- 702 Sam T Roweis and Lawrence K Saul. Nonlinear dimensionality reduction by locally linear embedding.  
703 *science*, 290(5500):2323–2326, 2000.  
704
- 705 Tim Sainburg, Leland McInnes, and Timothy Q Gentner. Parametric umap embeddings for represen-  
706 tation and semisupervised learning. *Neural Computation*, 33(11):2881–2907, 2021.  
707
- 708 Uri Shaham and Stefan Steinerberger. Stochastic neighbor embedding separates well-separated  
709 clusters. *arXiv preprint arXiv:1702.02670*, 2017.  
710
- 711 Karthik Shekhar, Sylvain W Lapan, Irene E Whitney, Nicholas M Tran, Evan Z Macosko, Monika  
712 Kowalczyk, Xian Adiconis, Joshua Z Levin, James Nemesh, Melissa Goldman, et al. Comprehen-  
713 sive classification of retinal bipolar neurons by single-cell transcriptomics. *Cell*, 166(5):1308–1323,  
714 2016.  
715
- 716 Xintong Shi, Wenzhi Cao, and Sebastian Raschka. Deep neural networks for rank-consistent ordinal  
717 regression based on conditional probabilities. *Pattern Analysis and Applications*, 26(3):941–955,  
718 2023.
- 719 Jian Tang, Jingzhou Liu, Ming Zhang, and Qiaozhu Mei. Visualizing large-scale and high-dimensional  
720 data. In *Proceedings of the 25th international conference on world wide web*, pp. 287–297, 2016.  
721
- 722 Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine*  
723 *learning research*, 9(11), 2008.  
724
- 725 Jarkko Venna and Samuel Kaski. Neighborhood preservation in nonlinear projection methods: An  
726 experimental study. In *International Conference on Artificial Neural Networks*, pp. 485–491.  
727 Springer, 2001.  
728
- 729 Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau,  
730 Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt,  
731 Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric  
732 Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas,  
733 Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris,  
734 Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0  
735 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature*  
736 *Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.
- 737 Yingfan Wang, Haiyang Huang, Cynthia Rudin, and Yaron Shaposhnik. Understanding how dimen-  
738 sion reduction tools work: an empirical approach to deciphering t-SNE, UMAP, TriMAP, and  
739 PaCMAP for data visualization. *The Journal of Machine Learning Research*, 22(1):9129–9201,  
740 2021.  
741
- 742 Yingfan Wang, Yiyang Sun, Haiyang Huang, and Cynthia Rudin. Dimension reduction with locally  
743 adjusted graphs. In *Thirty-Ninth AAAI Conference on Artificial Intelligence*, 2025.  
744
- 745 Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: A novel image dataset for bench-  
746 marking machine learning algorithms. 2017.  
747
- 748 Yang Yang, Hongjian Sun, Jialei Gong, Yali Du, and Di Yu. Interpretable dimensionality reduction  
749 by feature preserving manifold approximation and projection. *arXiv preprint arXiv:2211.09321*,  
750 2022.
- 751 Lihi Zelnik-Manor and Pietro Perona. Self-tuning spectral clustering. *Advances in neural information*  
752 *processing systems*, 17, 2004.  
753
- 754 Jonathan X Zheng, Samraat Pawar, and Dan FM Goodman. Graph drawing by stochastic gradient  
755 descent. *IEEE transactions on visualization and computer graphics*, 25(9):2738–2748, 2018.

756 APPENDIX  
757

758 In Appendix A, we provide necessary derivations regarding attraction and repulsion shapes of UMAP.  
759 In Appendix B we explore using a constant learning rate following our discussion in the main text.  
760 Then, in Appendix C, we discuss the implications of improved attraction and repulsion on large  
761 language model embeddings. We formally define Procrustes distance and matrix in Appendix D.1.  
762 We explore additional datasets for the random initialization experiment in Appendix E. We extend  
763 the discussion of NEG-*t*-SNE and UMAP in Appendix F. After that, in Appendix G, we explore  
764 alternate dimensionality reduction methods. Then, for completion, we discuss the construction of the  
765 high-dimensional graph in these methods in Appendix H. In Appendix I, we show the detailed result  
766 of varying  $\lambda_a$  and  $\lambda_r$  (from Fig. 1) for UMAP, NEG-*t*-SNE, and PaCMAP on different datasets. We  
767 show the effect of mixing and matching the shapes from different algorithms in Appendix J. Finally,  
768 we provide implementation details in Appendix K.

770 A PROOFS AND DERIVATIONS  
771

772 A.1 PROOF OF PROPOSITION 4.1  
773

774 *Proof.* We subtract Eq. (7) from Eq. (8) and take a norm:

$$775 \quad \|y_i^{t+1} - y_j^{t+1}\| = |1 + 2\lambda f_a| \|y_i^t - y_j^t\|. \quad (16)$$

776 This distance contracts as long as  $|1 + 2\lambda f_a| < 1$ , i.e., provided

$$777 \quad -1 < \lambda f_a < 0, \quad (17)$$

780 □

782 A.2 PROOF OF PROPOSITION 4.2  
783

784 *Proof.* We subtract Eq. (9) from Eq. (10) and take a norm:

$$785 \quad \|y_i^{t+1} - y_j^{t+1}\| = |1 + \lambda f_r| \|y_i^t - y_j^t\|. \quad (18)$$

786 This distance increases when  $|1 + \lambda f_r| > 1$ , i.e.,

$$787 \quad \lambda f_r < -2 \text{ or } f_r > 0. \quad (19)$$

788 From the definition of  $f_r$ , the latter suffices. □

792 A.3 ATTRACTION SHAPE:  
793

794 We use a general form of the low-dimensional affinity function, i.e.,  $q_{ij} = (\gamma + a\|y_i - y_j\|_2^b)^{-1}$ ,  
795 to derive the attraction shape. It reduces to UMAP for  $\gamma = 1$  and to NEG-*t*-SNE for  $\gamma = 2$ . The  
796 attractive force is given by

$$797 \quad \begin{aligned} 798 \quad \nabla_{y_i} \log q_{ij} &= -\nabla_{y_i} \log (\gamma + a(\|y_i - y_j\|_2^2)^b) \\ 799 \quad &= -\frac{1}{\gamma + a(\|y_i - y_j\|_2^2)^b} \nabla_{y_i} (\gamma + a(\|y_i - y_j\|_2^2)^b) \\ 800 \quad &= -\frac{1}{\gamma + a(\|y_i - y_j\|_2^2)^b} ab(\|y_i - y_j\|_2^2)^{b-1} \nabla_{y_i} \|y_i - y_j\|_2^2 \\ 801 \quad &= -\frac{2ab(\|y_i - y_j\|_2^2)^{b-1}}{\gamma + a(\|y_i - y_j\|_2^2)^b} (y_i - y_j). \end{aligned} \quad (20)$$

802 Defining  $\zeta = \|y_i - y_j\|_2$ , the first term gives the attraction shape as:

$$803 \quad f_a(\zeta) = -\frac{2ab\zeta^{2(b-1)}}{\gamma + a\zeta^{2b}} \quad (21)$$

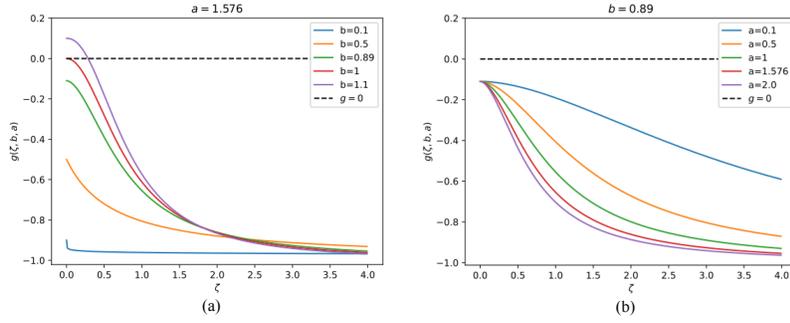
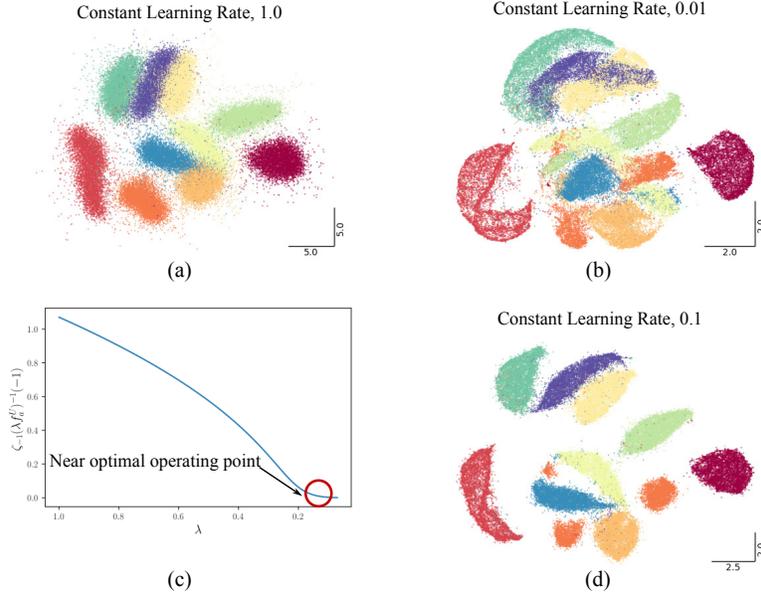
Figure 5: Values of  $g(\zeta, b, a)$  for (a)  $a$  fixed at 1.576 and (b)  $b$  fixed at 0.89.

Figure 6: Effect of constant learning rate in embeddings. (a) When the learning rate is too high ( $\lambda = 1.0$ ), the embeddings are diffuse (because of the high value of  $\zeta_{-1}$ ). (b) When the learning rate is too low ( $\lambda = 0.01$ ), clusters don't form (the strengths of attraction and repulsion are too low). (c)  $\zeta_{-1}$  decreases nonlinearly as the learning rate decreases. The goal of the algorithm is to reduce  $\zeta_{-1}$  while keeping effective levels of attraction and repulsion. (d) Distinct and compact clusters form at a constant, near-optimal learning rate  $\lambda = 0.1$ .

#### A.4 CONDITION FOR STRICTLY INCREASING $f_a^U$ :

Its behavior with distance can be characterized by computing the derivative of  $f_a^U$ :

$$\frac{df_a^U(\zeta)}{d\zeta} = -\frac{2ab\zeta^{2b-3}}{\gamma + a\zeta^{2b}} \left( (b-1) - \frac{ab\zeta^{2b}}{\gamma + a\zeta^{2b}} \right). \quad (22)$$

This leads to a strictly increasing condition ( $\frac{df_a^U}{d\zeta} > 0$ ),

$$g(\zeta, b, a) < 0, \quad (23)$$

where  $g(\zeta, b, a) = b - 1 - \frac{ab\zeta^{2b}}{\gamma + a\zeta^{2b}}$ . This inequality is valid as long as  $0 < b \leq 1$  (using the derivative and asymptotes of  $g$ ). Figure 5 shows values of  $g$  for different  $b$  and  $a$ . As  $b$  increases above 1, the inequality (23) no longer holds.

### 864 A.5 REPULSION SHAPE:

865 The repulsive force, using the general form of the low-dimensional affinity, is given by

$$\begin{aligned}
 866 \nabla_{y_i} \log(1 - q_{ij}) &= \nabla_{y_i} \log \left[ 1 - \frac{1}{\gamma + a(\|y_i - y_j\|_2^2)^b} \right] \\
 867 &= \frac{\gamma + a(\|y_i - y_j\|_2^2)^b}{(\gamma - 1) + a(\|y_i - y_j\|_2^2)^b} \nabla_{y_i} \left[ 1 - \frac{1}{\gamma + a(\|y_i - y_j\|_2^2)^b} \right] \\
 868 &= \frac{1}{\gamma - 1 + a(\|y_i - y_j\|_2^2)^b} \frac{ab(\|y_i - y_j\|_2^2)^{b-1}}{\gamma + a(\|y_i - y_j\|_2^2)^b} \nabla_{y_i} \|y_i - y_j\|_2^2 \\
 869 &= \frac{2ab(\|y_i - y_j\|_2^2)^{b-1}}{(\gamma - 1 + a(\|y_i - y_j\|_2^2)^b)(\gamma + a(\|y_i - y_j\|_2^2)^b)} (y_i - y_j). \quad (24)
 \end{aligned}$$

870 The first term gives the repulsion shape as:

$$871 f_r(\zeta = \|y_i - y_j\|_2) = \frac{2ab\zeta^{2(b-1)}}{(\gamma - 1 + a\zeta^{2b})(\gamma + a\zeta^{2b})}. \quad (25)$$

872 Generally,  $f_r > 0$ .

### 873 A.6 LOSS FUNCTIONS DUE TO MODIFIED ATTRACTION AND REPULSION SHAPE:

874 The cost function of the attractive term with the modification in Eq. (13) is given by

$$875 - \int (f_a^U(\|y_i - y_j\|_2) - \beta\|y_i - y_j\|_2)(y_i - y_j) dy_i = -\log(q_{ij}) + \frac{\beta}{3}\|y_i - y_j\|_2^3, \quad (26)$$

876 whereas the repulsive term due to Eq. (15) is given by

$$877 - \int (f_r^U(\|y_i - y_j\|_2) + \epsilon)(y_i - y_j) dy_i = -\log(1 - q_{ij}) - \frac{\epsilon}{2}\|y_i - y_j\|_2^2. \quad (27)$$

878 In both cases, the additional term acts as a regularizer, simply using norms. However, when we  
 879 directly add this term to attraction and repulsion shapes, we can easily explain what each term is  
 880 doing. For the attractive term, it is increasing far-sightedness, whereas for the repulsive term, it adds  
 881 a constant repulsive coefficient.

## 882 B MORE ON UMAP’S LEARNING RATE

883 As we discussed in Sections 5 and 6 of the main text, it is believed that UMAP requires learning  
 884 rate annealing (Fig 6). To explain this, in Section 5, we defined the concept of minimum distance  
 885 for contraction ( $\zeta_{-1}$ ) and established that reducing this value through learning rate annealing results  
 886 in compact clusters. Later, in Section 6, we compared attraction shapes of UMAP (for  $a = 1.0$  and  
 887  $b = 1.0$ ) and Neg- $t$ -SNE and explained that Neg- $t$ -SNE can withstand a constant learning rate of  
 888 1.0 better than UMAP because it’s attraction shape resides within  $[-1, 0]$  while UMAP’s is within  
 889  $[-2, 0]$ . Following the same logic, we showed that UMAP can withstand a constant learning of 0.5  
 890 by making its attraction shape stay within  $[-1, 0]$  (Figs. 11(f,h)).

891 However, the embeddings are still better if the learning rate anneals (for a wide range of parameters  
 892 of  $a$  and  $b$ ). This is because the goal of the algorithm is to eventually reduce  $\zeta_{-1}$  to zero and it occurs  
 893 when the learning rate reduces close to zero. Otherwise, the embedding becomes diffused (Fig 6(a)).  
 894 On the other hand, if the learning rate is too low, to begin with, the strength of attraction and repulsion  
 895 is too low, and thus no clear clusters can form (Fig 6(b)). By analyzing  $\zeta_{-1}$  vs  $\lambda$  curve (Fig 6(c)), we  
 896 see that a near optimal point is  $\lambda = 0.1$  where the value of  $\zeta_{-1}$  is low with considerable attraction  
 897 and repulsion strength. Setting the constant learning rate to 0.1, we obtain compact clusters with  
 898 clear boundaries (Fig 6(d)).

## 899 C LARGE LANGUAGE MODEL EMBEDDING

900 Increased attraction and repulsion often produce better embeddings. To show this, we randomly  
 901 selected 300,000 samples of features using PubMedBERT (Gu et al., 2021) with known labels

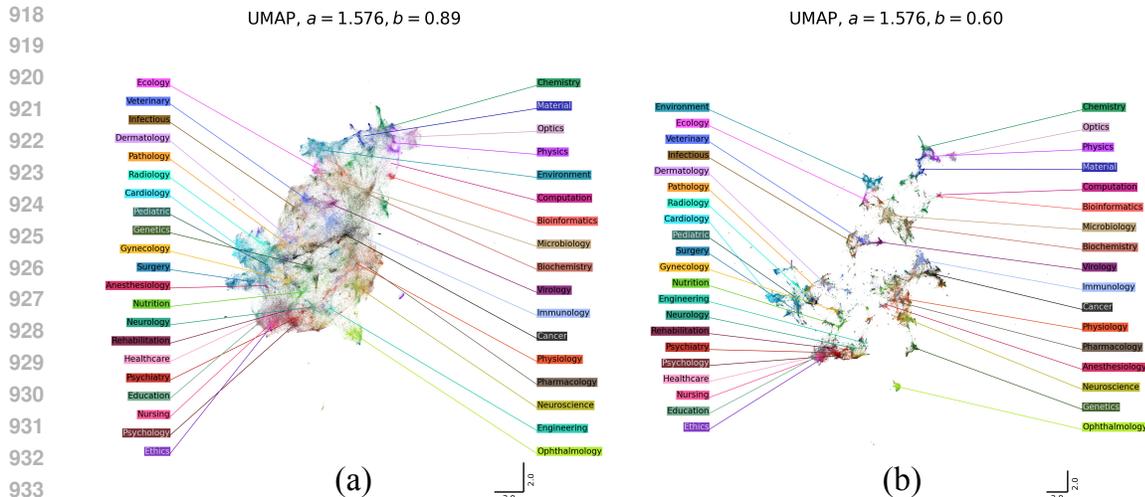


Figure 7: Visualization of 300K abstracts from the PubMed dataset. (a) Default UMAP ( $a = 1.576, b = 0.89$ ) shows labels overlapping each other in the mapping, but (b) UMAP with improved repulsion and attraction ( $a = 1.576, b = 0.60$ ) shows better cluster formation and non-overlapping labels. Both mappings are labeled for easier understanding.

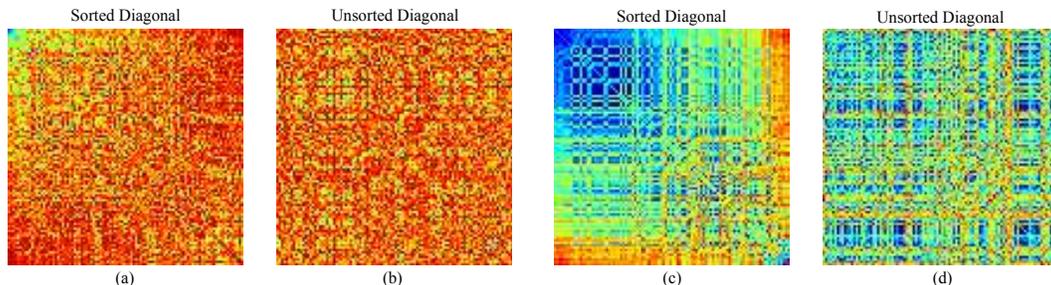


Figure 8: Effect of sorting the diagonal of Procrustes matrix on its visualization. (a) Procrustes matrix reproduced from 2(f) for the default UMAP attraction shape. The diagonal of the matrix is sorted by  $p_d$  of a sample embedding with PCA initialization. (b) The same data as in (a), but the diagonal is unsorted (or randomly sorted). (c) Procrustes matrix reproduced from 2(g) for modified attraction shape, where the diagonal of the matrix is sorted by  $p_d$  of a sample embedding with PCA initialization. (d) The same data as in (c), but the diagonal is unsorted.

from the PubMed dataset (González-Márquez et al., 2024). The default UMAP provides a crowded structure where labels (colors) overlap (Fig. 7(a)). Without labels, one might be perplexed about the embeddings, as explicit clusters are absent. However, by adjusting  $b$  to 0.6 (as discussed in Section 5.2), we increase repulsion and simultaneously facilitate cluster formation through improved attraction. The mapping is explorable even without explicit labeling (Fig. 7(b)). The improvement of the embedding can be quantified by the k-NN accuracy ( $k = 10$ ). On a hold-out data of 10,000 points, k-NN accuracy has increased from 49.2% to 55.48% by reducing  $b$  to 0.6.

## D METRICS

### D.1 PROCRUSTES DISTANCE AND PROCRUSTES MATRIX

The Procrustes distance (Gower, 1975) measures similarity between two point clouds  $\{x\}$  and  $\{y\}$  under linear transformations, viz. translation, scaling, and rotation. Operationally, we hold the former fixed and vary the latter until the two sets are in maximum alignment. Let  $\{y'\}$  be the transformation

of  $\{y\}$  that achieves this objective. Then the Procrustes distance is given by

$$p_d(\{x\}, \{y\}) = \sqrt{\sum_k (x_k - y'_k)^2}. \quad (28)$$

The Procrustes distance is a linear measure that has proven useful in a variety of settings (McInnes et al., 2018; Islam & Fleischer, 2022; Kotlov et al., 2024).

Here, we use the Procrustes distance to measure the consistency of embedding under random initialization. Let  $\{x\}_p$  be a reference embedding (using PCA initialization in our experiments), and  $X_r = \{\{x\}_i | i = 1, 2, 3, \dots, N\}$  be a set of  $N$  embeddings obtained from random initialization. The similarity of the embeddings can be quantified by taking a mean and standard deviation of the strictly lower triangular values of the matrix  $P$  (reported as mean $\pm$ std in Figs. 2, 9, and 10), with

$$\text{mean} = \frac{2}{N(N-1)} \sum_{i,j(i>j)} P_{i,j}, \text{ and } \text{std} = \sqrt{\sum_{i,j(i>j)} \frac{2(P_{i,j} - \text{mean})^2}{N(N-1)}} \quad (29)$$

The indexes of  $X_r$  can be sorted such that  $p_d(\{x\}_i, \{x\}_p) \leq p_d(\{x\}_{i+1}, \{x\}_p)$ , so that the diagonal values of the Procrustes matrix are given by

$$P_{i,i} = p_d(\{x\}_i, \{x\}_p) \quad (30)$$

and the off-diagonal values are given by

$$P_{i,j} = p_d(\{x\}_i, \{x\}_j). \quad (31)$$

Numerically, the sorting of the diagonal adds little value. Visually, however, the ordering reveals the underlying self-similarity of the embeddings. For example, in Fig. 8(a), the sorted diagonal shows that similar embeddings clump in the upper left region of the matrix. When we use a modified attraction shape, the number of points that are similar to each other increases (as shown by the larger blue region in Fig. 8(c)), indicating the presence of a metastable point in the embedding algorithm. On the other hand, when the diagonal is unsorted, this region disappears, and any sense of similar embeddings is lost (Fig. 8(b,d)).

## D.2 TRUSTWORTHINESS

The trustworthiness metric (Venna & Kaski, 2001) quantifies how well local neighborhoods are preserved after dimensionality reduction:

$$T = 1 - \frac{2}{nk(2n - 3k - 1)} \sum_{i=1}^n \sum_{y_j \in \text{KNN}(y_i, k)} \max(0, r(i, j) - k) \quad (32)$$

where  $\text{KNN}(y_i, k)$  is the  $k$ -NN graph in the embedding space and  $r(i, j)$  is the rank of  $x_j$  in the high-dimensional  $k$ -NN graph. In practice,  $k$  is often set to 5, assessing preservation of each point’s five nearest neighbors. For computational efficiency, when we report trustworthiness, we randomly sampled 10,000 indices. When comparing different embeddings, we used the same indices.

## D.3 SILHOUETTE SCORE

While the silhouette score (Rousseeuw, 1987) aims to assess clustering algorithms, we use it to evaluate label separation in the embeddings, i.e. how well the ground truth labels have been clustered. The idea is that the embedding algorithms naturally produce clusters and should separate the labels as much as possible. For a point  $y_i$  in a point cloud  $\{y\}$ , let  $a_i$  be the mean distance from  $y_i$  to other points in its own label, and let  $b_i$  be the minimal mean distance from  $y_i$  to points in any other label. The pointwise silhouette is thus given by

$$s_i = \frac{b_i - a_i}{\max\{a_i, b_i\}} \quad (33)$$

1026 This value lies within  $[-1, 1]$ . A value close to 1 means that  $y_i$  fits within its own label cluster, near 0  
1027 suggests a boundary point, and close to  $-1$  indicates failed label separation. The overall silhouette  
1028 score is

$$1029 \text{SIL} = \frac{1}{N} \sum s_i. \quad (34)$$

1030 We computed the silhouette score for the whole embedding (no random sampling) using Euclidean  
1031 distances.

## 1032 E EFFECT OF RANDOM INITIALIZATION IN ADDITIONAL DATASETS

1033 In the main text (Section 5.1), we showed results only on the MNIST dataset. Here we perform  
1034 the same experiment on the Fashion-MNIST (FMNIST) (Xiao et al., 2017) and single-cell tran-  
1035 scriptomes (Macosko et al., 2015) data (Fig. 9 and 10, respectively). The main conclusion remains  
1036 unchanged: modified and composite attraction shapes, such as those that increase attraction at large  
1037 distances, significantly improve the consistency of reconstruction.

1038  
1039  
1040  
1041  
1042  
1043  
1044  
1045  
1046  
1047  
1048  
1049  
1050  
1051  
1052  
1053  
1054  
1055  
1056  
1057  
1058  
1059  
1060  
1061  
1062  
1063  
1064  
1065  
1066  
1067  
1068  
1069  
1070  
1071  
1072  
1073  
1074  
1075  
1076  
1077  
1078  
1079

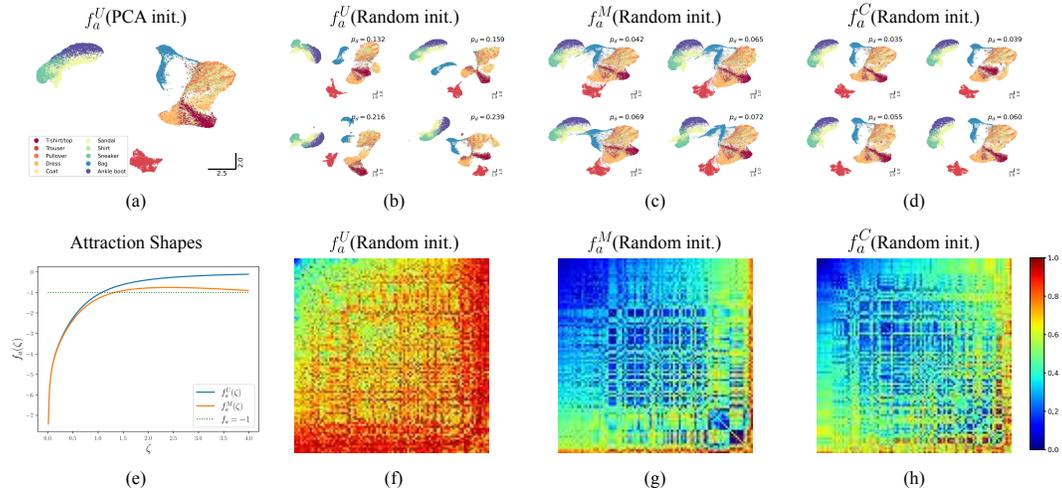


Figure 9: Effect of random UMAP initialization on different attraction shapes on FMNIST data. (a) Mapping using PCA as a standard. (b-d) Four mappings with the lowest Procrustes distance ( $p_d$ ) from the embedding in (a) for (b) default, (c) modified, and (d) composite attraction shapes. (e) Default UMAP and modified attraction shapes. (f-h) Procrustes matrix obtained from 100 runs of (f) default ( $0.72 \pm 0.15$ ), (g) modified ( $0.38 \pm 0.19$ ), and (h) composite ( $0.43 \pm 0.18$ ) attraction shapes.

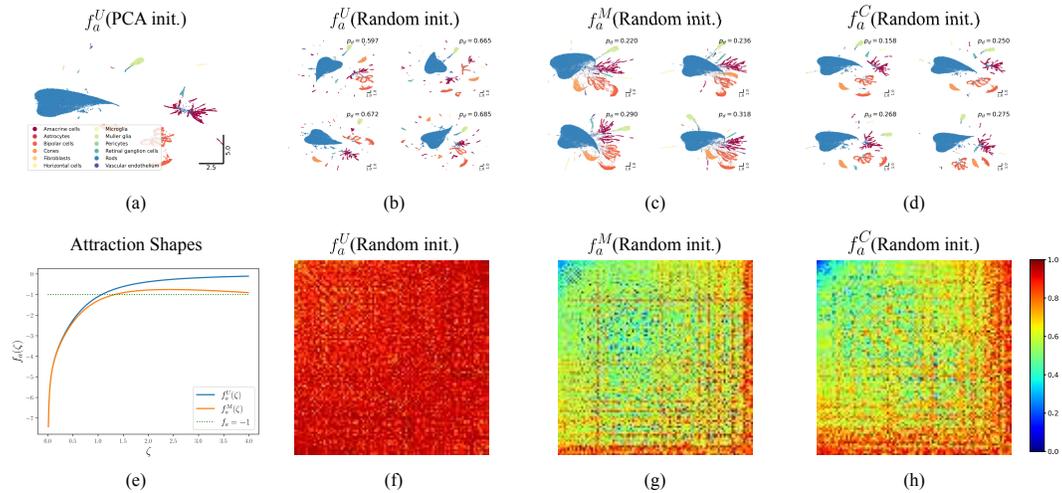


Figure 10: Effect of random UMAP initialization on different attraction shapes on single-cell transcriptomes data. (a) Mapping using PCA as a standard. (b-d) Four mappings with the lowest Procrustes distance ( $p_d$ ) from the embedding in (a) for (b) default, (c) modified, and (d) composite attraction shapes. (e) Default UMAP and modified attraction shapes. (f-h) Procrustes matrix obtained from 100 runs of (f) default ( $0.91 \pm 0.06$ ), (g) modified ( $0.61 \pm 0.13$ ), and (h) composite ( $0.64 \pm 0.15$ ) attraction shapes.

## F MORE ON NEG- $t$ -SNE

### F.1 COMPARISON TO UMAP

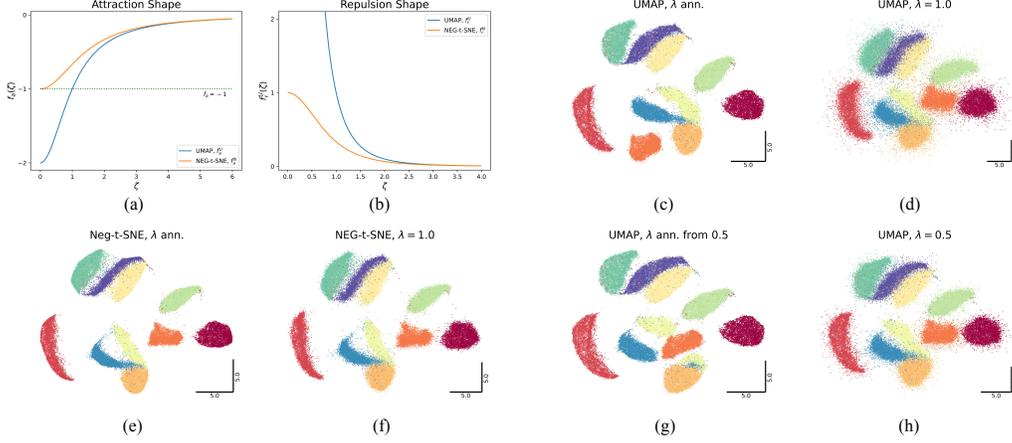


Figure 11: Sensitivity of UMAP and NEG- $t$ -SNE to learning rate on the MNIST dataset. (a) Attraction and (b) repulsion shapes for UMAP ( $a = 1$ ,  $b = 1$ ) and NEG- $t$ -SNE. (c,d) UMAP is very sensitive to the learning rate  $\lambda$ , as  $f_a^U < -1$  as the separation distance  $\zeta$  decreases. Thus, without annealing, the clusters become fuzzy. (e,f) NEG- $t$ -SNE is less sensitive to  $\lambda$  as  $f_a^N \in [-1, 0]$  always, and the clusters are thus less fuzzy even when not annealed. (g,h) Confining  $f_a^U$  to  $[-1, 0]$  by setting  $\lambda = 0.5$  shows less sensitivity to  $\lambda$ .

Figure 11 shows the shapes of UMAP and NEG- $t$ -SNE, along with various MNIST embeddings. When the learning rate is annealed, both UMAP and NEG- $t$ -SNE show similar output (Figs. 11(c,e)). However, when the learning rate is a constant value of 1, the UMAP shows a fuzzy structure, while NEG- $t$ -SNE shows a structure with much cleaner boundaries (Fig. 11(d,f)). The discussion in Section 6 suggests that constraining  $f_a^U$  within  $[-1, 0]$  can potentially result in less fuzzy clusters for fixed  $\lambda$ . We have seen this previously in Fig. 1(h,i) as well, where UMAP provided better embedding and clustering when the learning rate for attraction ( $\lambda_a$ ) was  $\lesssim 0.5$ . A straightforward way to achieve this is to initialize  $\lambda$  to 0.5, which satisfies Proposition 4.1 for all  $\zeta$ . The resulting embeddings, shown in Figs. 11 (g) and (h), confirm that clusters are similar to those of NEG- $t$ -SNE’s, and for a constant  $\lambda = 0.5$ , the clusters are less fuzzy than before as predicted with sharper boundaries (Fig. 11(d,h)). There are still a few points outside the clusters due to the characteristics of UMAP’s repulsion shape, which NEG- $t$ -SNE solves.

Next, we can introduce the parameters  $a$  and  $b$  into NEG- $t$ -SNE (essentially the formulation of Parametric UMAP; see Proposition 6.1). The affinity function becomes  $q_{ij}^N = 1/(2 + a\zeta^{2b})$ , and the attraction and repulsion shapes become  $f_a^N = -\frac{2ab\zeta^{2(b-1)}}{2+a\zeta^{2b}}$ , and  $f_r^N = \frac{2ab\zeta^{2(b-1)}}{(1+a\zeta^{2b})(2+a\zeta^{2b})}$ , respectively. For  $0 < b < 1$ , both shapes become unbounded as  $\zeta \rightarrow 0$ . Thus, NEG- $t$ -SNE will face similar numerical challenges to UMAP if  $a$  and  $b$  vary, and corresponding limitations carry over. One notable distinction is that, compared to UMAP, the attraction shape attains a lower minimum distance ( $\zeta_{-1}$ ) for the attraction. While this may enhance cluster formation, it approaches zero faster (increased near-sightedness as distance increases), potentially diminishing its effectiveness for attraction over longer distances.

### F.2 COMPARISON TO PARAMETRIC UMAP

Parametric UMAP was initially trained with the original UMAP objective (Sainburg et al., 2021), but later work adopted a numerically stable, log-sigmoid-based modified cross-entropy loss (Shi et al., 2023). This modification makes Parametric UMAP and Neg- $t$ -SNE Damrich et al. (2023) equivalent (Proposition 6.1). We show the equivalence below.

## PROOF OF PROPOSITION 6.1

*Proof.* The kernel function under this modification becomes

$$q_{ij}^P = -\log(1 + a\|y_i - y_j\|_2^{2b}). \quad (35)$$

The attractive term is

$$-\text{logsigmoid}(q_{ij}^P) = -\log\left(\frac{1}{1 + \exp(-q_{ij}^P)}\right) \quad (36)$$

$$= -\log\left(\frac{1}{2 + a\|y_i - y_j\|_2^{2b}}\right) \quad (37)$$

$$= -\log(q_{ij}^N). \quad (38)$$

And the repulsive term is

$$\text{logsidmoid}(q_{ij}^P) - q_{ij}^P = \log\left(\frac{1}{2 + a\|y_i - y_j\|_2^{2b}}\right) + \log(1 + a\|y_i - y_j\|_2^{2b}) \quad (39)$$

$$= \log\left(\frac{1 + a\|y_i - y_j\|_2^{2b}}{2 + a\|y_i - y_j\|_2^{2b}}\right) \quad (40)$$

$$= \log\left(1 - \frac{1}{2 + a\|y_i - y_j\|_2^{2b}}\right) \quad (41)$$

$$= \log(1 - q_{ij}^N). \quad (42)$$

Both these are NEG-t-SNE with explicit parameters  $a$  and  $b$  (while in Neg-t-SNE these are set to 1).  $\square$

## G ALTERNATE DIMENSIONALITY REDUCTION ALGORITHMS

The alternative algorithms we consider use the same kernel function as UMAP (with  $a = 1$  and  $b = 1$  in their low-dimensional weight):

$$q_{ij} = \frac{1}{1 + \|y_i - y_j\|_2^2}. \quad (43)$$

In this section, we first discuss the TriMap (Amid & Warmuth, 2019) algorithm. Even though, this algorithm relies on triplets (and not pairwise interactions), this works as a primer for analyzing attraction and repulsion that are a bit more involved than UMAP. This discussion is followed by Pairwise Controlled Manifold Approximation (PaCMAP) (Wang et al., 2021) and it’s extension Pairwise Controlled Manifold Approximation with Local Adjusted Graph (LocalMAP) (Wang et al., 2025) that modify TriMAP’s loss function that works for pairwise interactions. We then tackle  $t$ -SNE (Van der Maaten & Hinton, 2008). Then, we provide a short note on SNE (Hinton & Roweis, 2002) that uses an alternate kernel function and finally end the section by briefly discussing multidimensional scaling (Borg & Groenen, 2007).

## G.1 TRIMAP

TriMap (Amid & Warmuth, 2019) optimizes low-dimensional using a triplet loss

$$\mathcal{L}^T = \sum_{(i,j,k)} w_{ijk} \frac{1}{1 + \frac{q_{ij}}{q_{ik}}}, \quad (44)$$

where  $w_{ijk}$  is the weight of the triplet  $(y_i, y_j, y_k)$ ,  $y_j$  is in the  $k$ -nearest neighbor set of  $y_i$  in the high dimension, and  $y_k$  is a far-away point. When minimized, we expect that  $y_i$  and  $y_j$  attract each other, while  $y_i$  and  $y_k$  repel each other. The update equations are

$$y_i^{t+1} = y_i^t + \lambda f_a^T(\zeta_1^t, \zeta_2^t)(y_i^t - y_j^t) + \lambda f_r^T(\zeta_1, \zeta_2)(y_i^t - y_k^t), \quad (45)$$

$$y_j^{t+1} = y_j^t - \lambda f_a^T(\zeta_1^t, \zeta_2^t)(y_i^t - y_j^t), \quad (46)$$

$$y_k^{t+1} = y_k^t - \lambda f_r^T(\zeta_1^t, \zeta_2^t)(y_i^t - y_k^t), \quad (47)$$

1242  
 1243  
 1244  
 1245  
 1246  
 1247  
 1248  
 1249  
 1250  
 1251  
 1252  
 1253  
 1254  
 1255  
 1256  
 1257  
 1258  
 1259  
 1260  
 1261  
 1262  
 1263  
 1264  
 1265  
 1266  
 1267  
 1268  
 1269  
 1270  
 1271  
 1272  
 1273  
 1274  
 1275  
 1276  
 1277  
 1278  
 1279  
 1280  
 1281  
 1282  
 1283  
 1284  
 1285  
 1286  
 1287  
 1288  
 1289  
 1290  
 1291  
 1292  
 1293  
 1294  
 1295

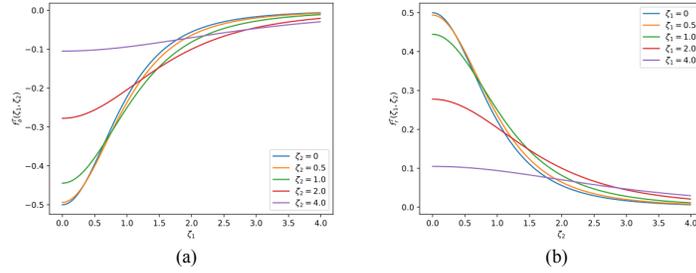


Figure 12: (a) Attraction shapes of TriMap for different  $\zeta_2$  and (b) repulsion shapes of TriMap for different  $\zeta_1$ .

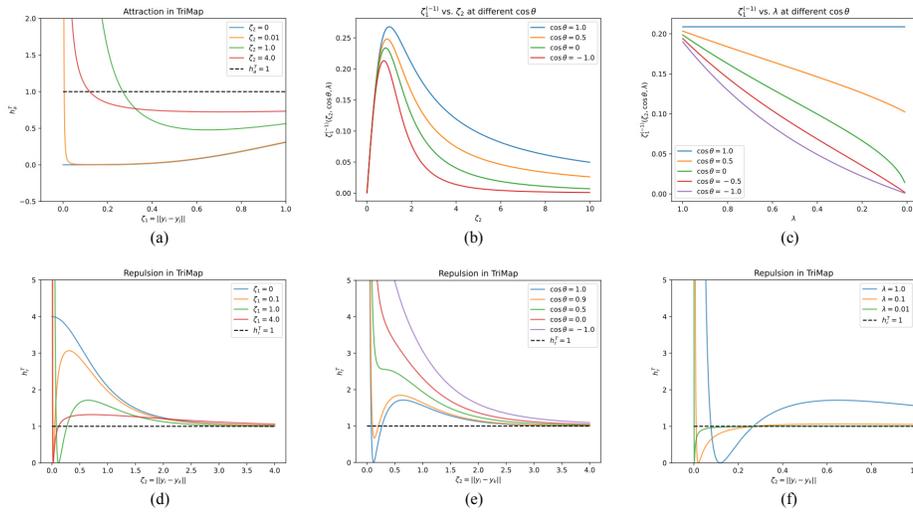


Figure 13: Attraction and repulsion behavior in TriMap. (a)  $h_a^T$  vs  $\zeta_1$  for different  $\zeta_2$ . Any values below the dotted line indicate attraction. Like UMAP, TriMAP shows attraction and repulsion for nearest neighbors ( $y_i, y_j$ ). (b,c) Unlike UMAP, the minimum distance for contraction ( $\zeta_1^{(-1)}$ ) varies due to dependence on (b)  $\zeta_2$  and (c)  $\lambda$ ; the function  $\cos \theta$  regulates the range of these values. (d-f) Repulsion in TriMap by varying (d)  $\zeta_1$ , (e)  $\cos \theta$ , and (f)  $\lambda$ . Values above (below) the dotted line indicate repulsion (attraction). While the repulsion force of UMAP shows only repulsion, that of TriMap can provide both attraction and repulsion. Unless otherwise labeled,  $\zeta_1 = 1.0$ ,  $\zeta_2 = 0.5$ ,  $\cos \theta = 1.0$ , and  $\lambda = 1.0$ .

where  $\zeta_1 = \|y_i - y_j\|_2$  is the distance between nearest neighbors,  $\zeta_2 = \|y_i - y_k\|_2$  is the distance from the faraway point, and  $f_a^T$  and  $f_r^T$  are attraction and repulsion shapes of TriMap, respectively. Unlike UMAP, the attractive and repulsive components are non-separable and the shapes depend on two distance measures (making them 2D). The functional form of the attraction shape is

$$f_a^T(\zeta_1, \zeta_2) = -\frac{2(1 + \zeta_2^2)}{(2 + \zeta_1^2 + \zeta_2^2)^2}, \quad (48)$$

and the repulsion shape is

$$f_r^T(\zeta_1, \zeta_2) = \frac{2(1 + \zeta_1^2)}{(2 + \zeta_1^2 + \zeta_2^2)^2}. \quad (49)$$

The attraction and repulsion shapes (Fig. 12) of TriMap shows similar trends of that of UMAP. However, the minimum value of repulsion shape is  $-0.5$ ; thus, unlike UMAP there is no position flipping in TriMAP due to attraction alone. However, since Eqs. (45-47) are not decoupled between attractive and repulsive terms, Propositions 4.1 and 4.2 do not apply. Focusing on attraction first, we show

**Proposition G.1.** *Update equations (45)-(47) provide a contraction if*

$$h_a^T(\zeta_1, \zeta_2, \theta, \lambda) < 1, \quad (50)$$

where  $h_a^T(\zeta_1, \zeta_2, \theta, \lambda) = (1 + 2\lambda f_a^T)^2 + 2(1 + 2\lambda f_a^T)\lambda f_r^T \frac{\zeta_2}{\zeta_1} \cos \theta + (\lambda f_r^T)^2 \frac{\zeta_2^2}{\zeta_1^2}$  and  $\theta$  is the angle between the vectors  $(y_i - y_j)$  and  $(y_i - y_k)$ , i.e.,  $\cos \theta = \frac{(y_i^t - y_j^t)^T (y_i^t - y_k^t)}{\|y_i^t - y_j^t\|_2 \|y_i^t - y_k^t\|_2}$ .

*Proof.* We require

$$\|y_i^{t+1} - y_j^{t+1}\|_2^2 < \|y_i^t - y_j^t\|_2^2. \quad (51)$$

From Eq. (45) and (46):  $y_i^{t+1} - y_j^{t+1} = (1 + 2\lambda f_a^T)(y_i^t - y_j^t) + \lambda f_r^T (y_i^t - y_k^t)$ . Putting this value in Eq. (51), we obtain the desired inequality.  $\square$

Inequality (50) depends on  $\zeta_1$ ,  $\zeta_2$ ,  $\cos \theta$  and  $\lambda$ . In particular, the value of  $\zeta_1 = \|y_i - y_j\|_2$ , where we want a contraction, is coupled with additional variables. Figure 13(a) shows the attraction behavior for various values of  $\zeta_2$ , while  $\cos \theta = 1$  and  $\lambda = 1$ . The values below the dotted line indicate attraction (and thus contraction of distance  $\zeta_1$ ), whereas the values above indicate repulsion (and therefore expansion of distance  $\zeta_1$ ). The value where the dotted line and  $h_a^T$  meet gives the minimum distance for contraction ( $\zeta_1^{(-1)}$ ) [analogous to  $\zeta_{-1}$  of UMAP], which we define as

$$\zeta_1^{(-1)}(\zeta_2, \theta, \lambda) = \arg \min_{\zeta_1} |h_a^T(\zeta_1, \zeta_2, \theta, \lambda) - 1|, \quad (52)$$

$$\text{s.t. } \zeta_1 \geq 0. \quad (53)$$

$\zeta_1^{(-1)}$  has a finite value and is  $> 0$  for most cases (Fig. 13(b,c)). As a result, TriMap can show behavior similar to UMAP and thus require learning rate annealing or a similar approach (in the TriMap implementation, the authors use the delta-bar-delta (Jacobs, 1988) method under appropriate initialization).

**Proposition G.2.** *Update equations (45)-(47) provide expansion if*

$$h_r^T(\zeta_1, \zeta_2, \theta, \lambda) > 1, \quad (54)$$

where  $h_r^T(\zeta_1, \zeta_2, \theta, \lambda) = (1 + 2\lambda f_r^T)^2 + 2\lambda f_a^T (1 + \lambda f_r^T) \frac{\zeta_1}{\zeta_2} \cos \theta + (\lambda f_a^T)^2 \frac{\zeta_1^2}{\zeta_2^2}$ .

*Proof.* We require

$$\|y_i^{t+1} - y_k^{t+1}\|_2^2 > \|y_i^t - y_k^t\|_2^2. \quad (55)$$

From Eq. (45) and (47):  $y_i^{t+1} - y_k^{t+1} = (1 + 2\lambda f_r^T)(y_i^t - y_k^t) + \lambda f_a^T (y_i^t - y_j^t)$ . Putting this value in Eq. (55) we obtain inequality (54).  $\square$

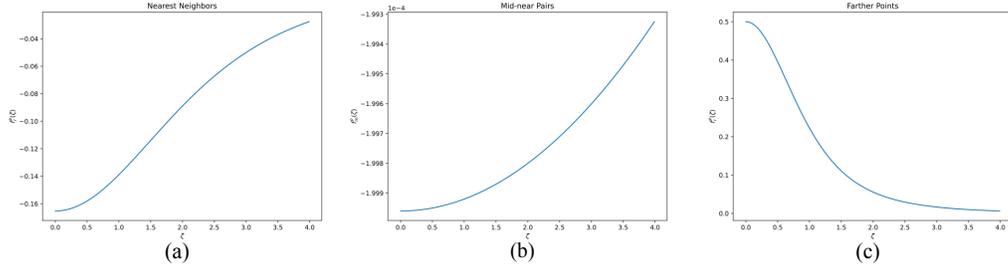


Figure 14: Attraction and repulsion shapes of PaCMAP. (a,b) Attraction shapes for (a) nearest-neighbor and (b) mid-near points (note that the values are on the order of  $10^{-4}$ ). (c) Repulsion shape for farther pairs.  $\lambda = 1$  for all figures.

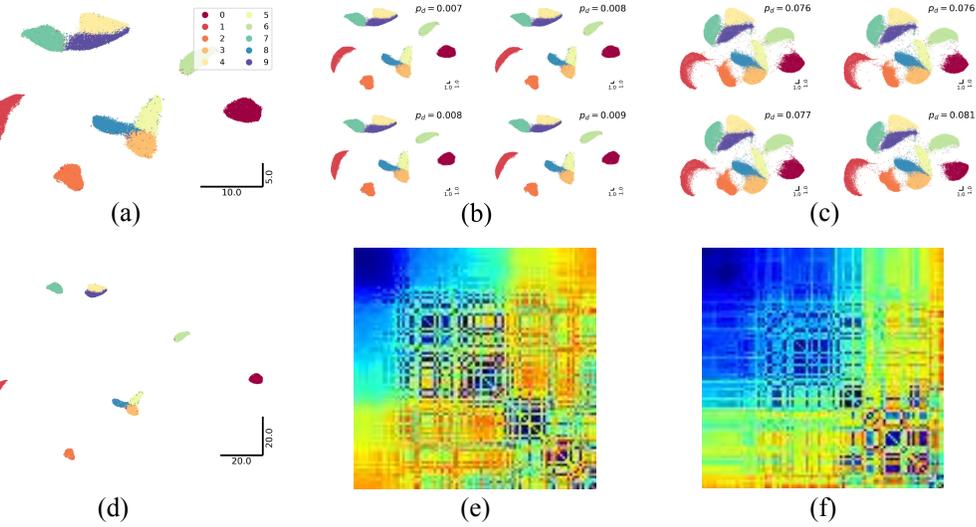


Figure 15: PaCMAP behavior for different conditions. (a) PaCMAP of MNIST data with PCA initialization. (b,c) Four samples that best match with (a) by (b) original PaCMAP and (c) PaCMAP with modified attraction shape,  $f_a^M(\zeta) = f_a^P(\zeta) - 0.001\zeta$ , when randomly initialized. (d) PaCMAP of MNIST with a modified repulsion shape  $f_r^M = f_r^P + 0.00005$ . (e,f) Procrustes matrix for (e) original PaCMAP ( $0.51 \pm 0.22$ ) and (f) PaCMAP with modified attraction shape ( $0.43 \pm 0.22$ ). Similar to UMAP, increased attraction at farther distances show improved consistency, while increased repulsion shows smaller clusters and larger inter-cluster distances.

Inequality (54) also depends on the set  $\zeta_1, \zeta_2, \cos \theta$  and  $\lambda$ . Here, we are interested in the expansion of  $\zeta_2 = \|y_i - y_k\|_2$ . Figures 13(d-f) show the repulsion behavior by varying the other quantities. Any values above the dotted line indicate repulsion (and thus expansion of distance), while the values below indicate attraction. The striking difference compared to UMAP is that repulsion in TriMap can cause contraction instead of expansion. Since this anomaly occurs for small distances, it can be avoided by an appropriate initialization and choice of triplets.

## G.2 PAIRWISE CONTROLLED MANIFOLD APPROXIMATION (PACMAP)

PaCMAP (Wang et al., 2021) optimizes low-dimensional embedding at different scales. The loss function is

$$\mathcal{L}^P = w_{NB} \sum_{(i,j) \in NN} \frac{1}{1 + 10q_{ij}} + w_{MN} \sum_{(i,k) \in MN} \frac{1}{1 + 10000q_{ik}} + w_{FP} \sum_{(i,l) \in FP} \frac{1}{1 + \frac{1}{q_{il}}}, \quad (56)$$

where  $w_{NB}$ ,  $w_{MN}$ , and  $w_{FP}$  are weights of the nearest neighbor (NN) pairs, mid-near (MN) pairs, and further pairs (FP), respectively (details in Appendix H). The first two terms provide attraction, whereas the last term provides repulsion. A closer look at the loss function reveals that the function is a modified form of TriMap’s triplet loss. For the attractive terms, it replaces TriMap’s affinity for distant points with constant terms (1/10 for nearest neighbors, 1/10000 for mid-near pairs, and 1 for farther points). This loss function is thus separable into three terms and decouples the update equations. The update equations of the nearest neighbor term are

$$y_i^{t+1} = y_i^t + \lambda f_a^P(\zeta_1^t)(y_i^t - y_j^t), \quad (57)$$

$$y_j^{t+1} = y_j^t - \lambda f_a^P(\zeta_1^t)(y_i^t - y_j^t), \quad (58)$$

of the mid-near pairs are

$$y_i^{t+1} = y_i^t + \lambda f_m^P(\zeta_2^t)(y_i^t - y_k^t), \quad (59)$$

$$y_k^{t+1} = y_k^t - \lambda f_m^P(\zeta_2^t)(y_i^t - y_k^t), \quad (60)$$

and of the farthest pairs are

$$y_i^{t+1} = y_i^t + \lambda f_r^P(\zeta_3^t)(y_i^t - y_l^t), \quad (61)$$

$$y_l^{t+1} = y_l^t - \lambda f_r^P(\zeta_3^t)(y_i^t - y_l^t), \quad (62)$$

where  $\zeta_1 = \|y_i - y_j\|_2$ ,  $\zeta_2 = \|y_i - y_k\|_2$ ,  $\zeta_3 = \|y_i - y_l\|_2$  are distances,  $f_a^P$  and  $f_m^P$  are attraction shapes for nearest neighbors and mid-near pairs, respectively, and  $f_r^P$  is the repulsion shape for the farthest pairs. Correspondingly, the functional forms of the shapes are

$$f_a^P(\zeta) = -\frac{20}{(11 + \zeta^2)^2}, \quad (63)$$

$$f_m^P(\zeta) = -\frac{20000}{(10001 + \zeta^2)^2}, \quad (64)$$

$$f_r^P(\zeta) = \frac{2}{(2 + \zeta^2)^2}. \quad (65)$$

$f_a^P$  and  $f_m^P$  follow Proposition 4.1, and  $f_r$  follows Proposition 4.2 (Fig. 14). The attraction is quite low compared to UMAP, but it is good enough for a wide range of learning rates (modulated by the Adam algorithm (Kingma & Ba, 2015)); with  $w_{NB} = 3$  the maximum attraction is always below 0.5 preventing any flips during attractin update. Typically, PaCMAP initializes the embedding within a small sphere in the low dimension (e.g., in (Wang et al., 2021), the initialization is often on the order of  $10^{-3}$ ) and relies on repulsion to separate the individual clusters. Overall, it mostly recovers UMAP’s clustering properties (especially for MNIST) with improved ordering. The consistency under random initialization is better than UMAP (Fig. 15(b,e)) which can be improved further using a modified attraction (Fig. 15(c,f)). Increasing repulsion by adding a small value to the repulsion shape increases compactness of the embedding (by increasing inter-cluster distance).

### G.3 PAIRWISE CONTROLLED MANIFOLD APPROXIMATION WITH LOCAL ADJUSTED GRAPH (LOCALMAP)

LocalMAP (Wang et al., 2025) is an iteration of the PaCMAP algorithm. One of the defining features of LocalMAP is the separation of all 10 clusters of the MNIST data (Fig. 16(a)), with behavior similar to the ones in Figs. 3(g,h,j,k) and Figs. 4(a,b). Here, we explore the interplay of attractive and repulsive forces on the compactness and connectedness of clusters.

LocalMAP performs PaCMAP and then does additional optimization on the attraction to decouple some clusters. To this end, it minimizes the following loss function

$$\mathcal{L}^L = \sum_{(i,j) \in NN} \frac{K}{\frac{1}{\sqrt{q_{ij}}} + C\sqrt{q_{ij}}} + \sum_{(i,l) \in FP} \frac{1}{1 + \frac{1}{q_{il}}}. \quad (66)$$

The first term amalgamates the attractive and repulsive nature of the triplet loss function that works on the same pair. In one regime, this function causes attraction, while in the other, it causes repulsion. The second term is identical to the ones in PaCMAP; the only difference is that the algorithm

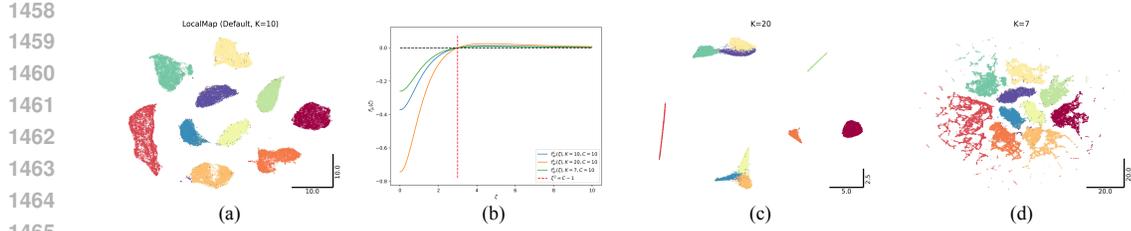


Figure 16: Behavior of LocalMAP on MNIST data. (a) Default embedding. (b) Attraction-repulsion shape of nearest neighbors. A value below (above) the dotted line indicates attraction (repulsion). Transition from attraction to repulsion occurs as  $\zeta$  increases and crosses  $\sqrt{C-1}$ . Following implementation of LocalMAP, we used  $C = 10$ . (c) When  $K$  is large ( $= 20$ ), repulsion dominates and clusters become compact. (d) When  $K$  is small ( $= 7$ ), attraction dominates and clusters break up.

resamples further pairs every few iterations. Thus, we analyze only the first term involving nearest neighbors. The update equations are

$$y_i^{t+1} = y_i^t + \lambda f_{ar}^L(\zeta_1^t)(y_i^t - y_j^t), \quad (67)$$

$$y_j^{t+1} = y_j^t - \lambda f_{ar}^L(\zeta_1^t)(y_i^t - y_j^t), \quad (68)$$

where  $f_{ar}^L$  is the attraction-repulsion shape given by

$$f_{ar}^L(\zeta) = -\frac{K(C-1-\zeta^2)}{2\sqrt{1+\zeta^2}(1+C+\zeta^2)^2}. \quad (69)$$

The update provides contraction as long as  $\zeta^2 < C-1$  and  $-1 < \lambda f_{ar}^L < 0$  (from Proposition 4.1). When  $\zeta^2 > C-1$ ,  $f_{ar} > 0$ , and by Proposition 4.2 the update equations provide expansion. The values of  $K$  and  $C$  determine whether attraction or repulsion dominates the dynamics. In LocalMAP implementation, both  $C$  and  $K$  are set to 10 (Fig. 16(b)), and the strength of attraction is higher than PaCMAP (Fig. 14(a)). As a result, when  $\zeta > 3$ , the nearest neighbors face repulsion, causing pairs bridging two clusters to separate. The value of  $K$ , working as a scaling parameter for the forces, regulates this separation.

Using Proposition 4.1,  $\lambda f_{ar}^L(0) \geq -\frac{1}{2}$  gives the maximum values of  $K$ , and the maximum attraction possible by  $f_{ar}^L$ , without flipping the placements of the pairs. (We would want to avoid flipping the pairs at the LocalMAP optimization to preserve the ordering from PaCMAP; otherwise, it may inhibit cluster separation.) This simplifies (69) to  $K \leq \frac{\lambda(1+C)^2}{C-1}$ , which for  $C = 10$  and  $\lambda = 1$  gives  $K \lesssim 13.44$ . Moreover, at a higher value of  $K \gtrsim 13.44$ , the repulsive forces dominate, and the clusters become more compact, but objective of LocalMAP fails (the bridge between clusters persists in Fig. 16(c) for  $K = 20$ ). On the other hand, as  $K$  decreases, attractive forces dominate (because repulsive forces are too low), and the embedding shows the breaking up of existing clusters (Fig. 16(d) for  $K = 7$ , which mimics the one in Fig. 4(b)).

#### G.4 $t$ -DISTRIBUTED STOCHASTIC NEIGHBOR EMBEDDING ( $t$ -SNE)

$t$ -SNE (Van der Maaten & Hinton, 2008) optimizes pairwise distances. The loss function is

$$\mathcal{L} = -\sum_{i,j} w_{i,j} \log \left( \frac{q_{i,j}}{\sum_{k \neq l} q_{k,l}} \right), \quad (70)$$

where  $w_{i,j}$  is the weight of the pair. The original implementation of  $t$ -SNE considers all the pairs (not just nearest neighbors). This loss function decomposes into attraction and repulsion forces by

$$\mathcal{L} = \sum_{i,j} \left[ -w_{i,j} \log(q_{i,j}) + w_{i,j} \log \left( \sum_{k \neq l} q_{k,l} \right) \right]. \quad (71)$$

As previously, the first term provides the attractive forces and the second term provides the repulsive forces. While the attractive term is identical to that of UMAP (with  $a = 1$  and  $b = 1$ ) and is easy to

compute, the repulsive term is coupled among every pair and thus, is very costly. Using the same principles we applied for UMAP, we can write the update equations of  $t$ -SNE. Since the original  $t$ -SNE didn't rely on the nearest neighbor graph, the weight  $w_{i,j}$ , computed for all the pairs, is important in the update equations. The attractive update equations are

$$y_i^{t+1} = y_i^t + \lambda w_{i,j} f_a^{t-SNE}(\zeta_{i,j}^t)(y_i^t - y_j^t), \quad (72)$$

$$y_j^{t+1} = y_j^t - \lambda w_{i,j} f_a^{t-SNE}(\zeta_{i,j}^t)(y_i^t - y_j^t), \quad (73)$$

where  $f_a^{t-SNE}$  is the attraction shape of  $t$ -SNE and  $\zeta_{i,j} = \|y_i - y_j\|_2$ . The update equation for the repulsive parts are

$$y_i^{t+1} = y_i^t + \lambda \frac{w_{i,j}}{Z} \sum_k f_r^{t-SNE}(\zeta_{i,k}^t)(y_i^t - y_k^t), \quad (74)$$

$$y_j^{t+1} = y_j^t - \lambda \frac{w_{i,j}}{Z} \sum_l f_r^{t-SNE}(\zeta_{l,j}^t)(y_l^t - y_j^t), \forall j, j \neq i, \quad (75)$$

where  $f_r^{t-SNE}$  is the repulsion shape of  $t$ -SNE and  $Z = \sum_{k \neq l} q_{k,l}$ . The functional forms of these shapes are

$$f_a^{t-SNE}(\zeta) = -\frac{2}{1 + \zeta^2}, \text{ and} \quad (76)$$

$$f_r^{t-SNE}(\zeta) = \frac{2}{(1 + \zeta^2)^2}. \quad (77)$$

From the attractive update Eqs. (72)-(73),  $f_a^{t-SNE}$  follows Proposition 4.1 (with  $0 < \lambda w_{i,j} f_a^{t-SNE} < -1$ ) and gives the minimum distance for contraction,  $\zeta_{-1}$ ). The repulsive update is coupled with all the pairs and thus does not have a simple relation to the repulsion shape. Rather, enabled by our experience from TriMap's analysis, we can derive the following:

**Proposition G.3.** *The update Eqs. (74)-(75) provide an expansion if*

$$h_r^{t-SNE}(\zeta_{i,j}, v, \theta, \lambda, w_{i,j}) > 1, \quad (78)$$

where

$$h_r^{t-SNE}(\zeta_{i,j}, v, \theta, \lambda, w_{i,j}) = (1 + 2\lambda \frac{w_{i,j}}{Z} f_r^{t-SNE}(\zeta_{i,j}))^2 + \frac{\|v\|_2^2}{\zeta_{i,j}^2} + 2\lambda \frac{w_{i,j}}{Z} f_r^{t-SNE}(\zeta_{i,j}) \frac{\|v\|_2}{\zeta_{i,j}} \cos \theta,$$

$$v = \lambda \frac{w_{i,j}}{Z} \left( \sum_{k, k \neq j} f_r^{t-SNE}(\zeta_{i,k}) (y_i - y_k) + \sum_{l, l \neq i} f_r^{t-SNE}(\zeta_{l,j}) (y_l - y_j) \right),$$

and  $\theta$  is the angle between  $(y_i - y_j)$  and  $v$ .

*Proof.* We require,

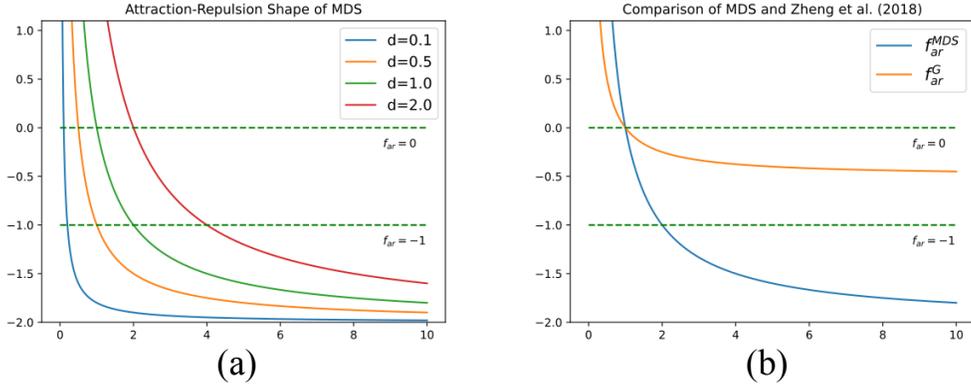
$$\|y_i^{t+1} - y_j^{t+1}\|_2^2 > \|y_i^t - y_j^t\|_2^2. \quad (79)$$

From Eqs. (74) and (75),  $y_i - y_j = (1 + 2\lambda \frac{w_{i,j}}{Z} f_r^{t-SNE}(\zeta_{i,j}))(y_i - y_j) + v$ . Putting this value in Eq. (79), we obtain the desired inequality.  $\square$

Since  $t$ -SNE's condition for repulsion (Eq. 78) resembles that of TriMap, the repulsion behavior will be the same. Thus,  $t$ -SNE's repulsive forces can give both attraction and repulsion.

Since,  $t$ -SNE's forces are scaled (by  $w_{i,j}$  for attraction and  $\frac{w_{i,j}}{Z}$  for repulsion), the attractive and repulsive forces are typically lower than that of UMAP. As a result, the algorithm generally uses large values for learning rate (e.g.,  $\approx 10^3$  in the Open- $t$ -SNE package (Poličar et al., 2024)). Moreover, most  $t$ -SNE implementations require an 'early exaggeration' step where the attractive forces are multiplied by a constant value for the first few iterations. This causes points that are supposed to be closer but currently placed far apart to approach each other (inducing far-sightedness). On the other hand, if some points are very close ( $\zeta_{i,j} \approx \zeta_{-1}$ ) but require separation, this early exaggeration trick achieves that as well. Thus, 'early exaggeration' plays a vital role in finding a consistent embedding in  $t$ -SNE and is an indispensable feature of the algorithm; especially when initialized randomly. (This trick, when applied throughout the optimization, makes  $t$ -SNE embeddings look closer to UMAP embeddings (Böhm et al., 2022).)

1566  
1567  
1568  
1569  
1570  
1571  
1572  
1573  
1574  
1575  
1576  
1577



1578  
1579  
1580  
1581  
1582

Figure 17: (a) Attraction-repulsion characteristic of MDS using  $f_{ar}$ . (b) Comparison of attraction-repulsion characteristic of MDS and the method in (Zheng et al., 2018) for  $d = 1.0$ .

1583  
1584

### G.5 STOCHASTIC NEIGHBOR EMBEDDING (SNE)

1585  
1586  
1587

SNE (Hinton & Roweis, 2002) is one of the oldest algorithms in this class. This follows the same formula of  $t$ -SNE (Eq. 71), but with  $q_{i,j} = \exp(-\|y_i - y_j\|_2^2)$ . This results in the attraction shape

1588

$$f_a^{SNE}(\zeta) = -2, \quad (80)$$

1589

that follows Proposition 4.1 (with  $0 < \lambda w_{i,j} f_a^{SNE} < -1$ ) and the repulsion shape

1590

$$f_r^{SNE}(\zeta) = 2 \exp(-\zeta^2), \quad (81)$$

1591

1592

that follows proposition G.3 (by replacing the  $t$ -SNE symbols with SNE counterparts). The attraction shape is ill-posed and thus mainly relies on the values of learning rate ( $\lambda$ ) and the weights ( $w_{i,j}$ ) for contraction resulting in clusters overlapping each other even for small number of samples (called crowding problem (Van der Maaten & Hinton, 2008)), which  $t$ -SNE and the subsequent algorithms improve by moving away from the Gaussian kernel to a heavy-tailed one, i.e., Eq. (43).

1593

1594

1595

1596

1597

1598

1599

### G.6 MULTIDIMENSIONAL SCALING (MDS)

1600

1601

1602

1603

1604

1605

Multidimensional scaling (MDS) (Borg & Groenen, 2007) typically does not use gradient methods, as they often fail to converge to good mappings; instead, it employs stress majorization. Nevertheless, few works discuss gradient methods (Kruskal, 1964; Zheng et al., 2018). We offer a brief treatment for this below. Particularly, Zheng et al. (2018) formulates a successful gradient descent-based MDS algorithm for graph drawing. We start with the cost function (we can ignore the weight  $w_{i,j}$  without loss of generality to their approach, and for discussion, we can lump it into the learning rate):

1606

1607

1608

$$\mathcal{L}^{MDS} = \sum_{i,j,i < j} (\|y_i - y_j\| - d_{ij})^2. \quad (82)$$

1609

1610

1611

The loss has a singleton term for each pair and does not have explicit attractive and repulsive terms. The term itself will provide both attraction and repulsion. Fortunately, the loss is separable into individual  $(i, j)$  components, allowing us to analyze a single pair. The derivative is given by

1612

1613

1614

$$\frac{\partial}{\partial y_i} (\|y_i - y_j\| - d_{ij})^2 = 2 \frac{\|y_i - y_j\| - d_{ij}}{\|y_i - y_j\|} (y_i - y_j). \quad (83)$$

1615

Thus, the attraction-repulsion shape of MDS becomes

1616

1617

1618

$$f_{ar}^{MDS}(\zeta) = -2 \frac{\zeta - d}{\zeta}, \quad (84)$$

1619

where  $\zeta = \|y_i - y_j\|_2$ . Here, the shape is attractive when  $\zeta > d$ . In this range, the shape is bounded within  $[-2, 0]$  (Fig. 17(a)). However, this is not suitable for convergence as values only within  $[-1, 0]$

contract, while the others expand. To make things worse, values lower than  $-2$  cause the points to flip and expand. To work around this, Zheng et al. (2018) uses an ad-hoc formulation inspired by the force-directed graphs, given by

$$f_{ar}^G(\zeta) = -\frac{1}{2} \frac{\zeta - d}{\zeta}, \quad (85)$$

which works for an effective learning rate  $\lambda \leq 1$ .  $f_{ar}^G$  is bounded within  $[-0.5, 0]$  and thus,  $\lambda \leq 1$  works. On the other hand, for  $\zeta < d$ ,  $f_{ar}^{MDS}(> 0)$  shows repulsive behavior. Overall, this attraction-repulsion interaction works best when the initialization is already close to a desired output. If one keeps optimizing for a pair, it will oscillate around the distance ( $\zeta = d$ ) where  $f_{ar}^G = 0$  (and consequently, we can have a notion of distance similar to  $\zeta_{-1}$  of UMAP). No choice of learning rate reduces this distance to zero (in fact, this achieves the objective of MDS). Thus, the clusters are often fuzzy compared to methods like UMAP and PaCMAP (for relevant illustrations, see Lambert et al. (2022), de Bodt et al. (2025), and Kury et al. (2025)). Note that Lambert et al. (2022) converts the MDS loss function to a quartet stress (loss involving four samples) and a relative distance (distance divided by the sum of six distances in the quartet), enabling global regularization and reduced computation than the majorization approach.

## H CONSTRUCTION OF THE HIGH-DIMENSIONAL GRAPH

Stochastic Neighbor Embedding (SNE) Hinton & Roweis (2002) underpins modern dimensionality reduction algorithms. It constructs a high-dimensional graph of the dataset  $X = \{x_i \in R^n | i = 1, \dots, N\}$  by the following system of equations:

$$w_{ij} = \frac{w_{j|i} + w_{i|j}}{2N}, \quad (86)$$

$$w_{j|i} = \frac{\exp(-\|x_i - x_j\|_2^2 / 2\sigma_i^2)}{\sum_{t \neq v} \exp(-\|x_t - x_v\|_2^2 / 2\sigma_t^2)}, \quad (87)$$

where  $\sigma_i^2$  is chosen to match a user-defined value *perplexity*,  $P$ , defined as

$$P = 2^{H_i} \quad (88)$$

$$H_i = \sum_{j \neq i} w_{j|i} \log_2 w_{j|i}. \quad (89)$$

On the other hand, UMAP constructs its high-dimensional graph by the following system of equations relying on the k-nearest neighbor (k-NN) algorithm:

$$p_{i|j} = \begin{cases} \exp\left(-\frac{d(x_i, x_j) - \rho_i}{\sigma_i}\right) & \text{if } x_j \in \text{KNN}(x_i, k), \\ 0 & \text{otherwise} \end{cases}, \quad (90)$$

$$\rho_i = \min_{x_j \in \text{KNN}(x_i, k)} d(x_i, x_j), \quad (91)$$

where  $\text{KNN}(x_i, k)$  is the set of  $k$ -nearest neighbors of the point  $x_i$  and  $\sigma_i$  is a scaling parameter such that  $\sum_j p_{i|j} = \log_2(k)$ . The graph is then symmetrized by a t-conorm:

$$p_{i,j} = p_{i|j} + p_{j|i} - p_{i|j}p_{j|i}. \quad (92)$$

PaCMAP uses just the k-NN graph for the affinities (all equal to 1) with a self-tuning distance measure (Zelnik-Manor & Perona, 2004),

$$d_{i,j}^2 = \frac{\|x_i - x_j\|^2}{\sigma_i \sigma_j}, \quad (93)$$

where  $\sigma_i$  is the average distance between  $x_i$  and its Euclidean nearest fourth to sixth neighbors. The purpose of this is the same as the corresponding  $\sigma_i$  parameters in Eq. (87) and (90), despite all three being defined differently.

Regardless of these choices, the optimization, and by our analysis, the attraction and repulsion shapes are the primary drivers of the low-dimensional embedding.

## I DETAILED RESULTS FOR VARYING $\lambda_a$ AND $\lambda_b$

In this section, we provide detailed results for the experiments in Fig. 1((h,i)). We obtained these results, by changing the attraction and repulsion shapes of UMAP to that of the other methods. Figure 18 reproduces the results given in the main text. Figures 19-22 show the individual embeddings for each of the choices of  $\lambda_a$  and  $\lambda_r$  for each methods along with their initialization, a reference embedding when learning rate,  $\lambda$ , is annealed, and when either the  $\lambda_a$  or  $\lambda_r$  set to 0. For PaCMAP, which uses the concept of mid-near pairs, we show additional reference of mid-near pairs included as well. Figures 23- 27 and Figures. 28- 32 provide results on FMNIST and single-cell transcriptomes dataset, respectively.

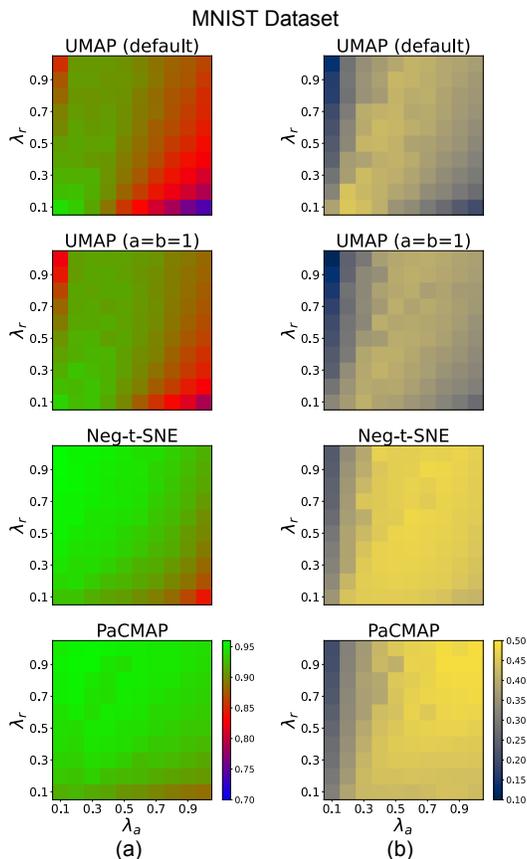


Figure 18: (a) Trustworthiness and (b) Silhouette score of different methods for the MNIST dataset.

1728  
 1729  
 1730  
 1731  
 1732  
 1733  
 1734  
 1735  
 1736  
 1737  
 1738  
 1739  
 1740  
 1741  
 1742  
 1743  
 1744  
 1745  
 1746  
 1747  
 1748  
 1749  
 1750  
 1751  
 1752  
 1753  
 1754  
 1755  
 1756  
 1757  
 1758  
 1759  
 1760  
 1761  
 1762  
 1763  
 1764  
 1765  
 1766  
 1767  
 1768  
 1769  
 1770  
 1771  
 1772  
 1773  
 1774  
 1775  
 1776  
 1777  
 1778  
 1779  
 1780  
 1781

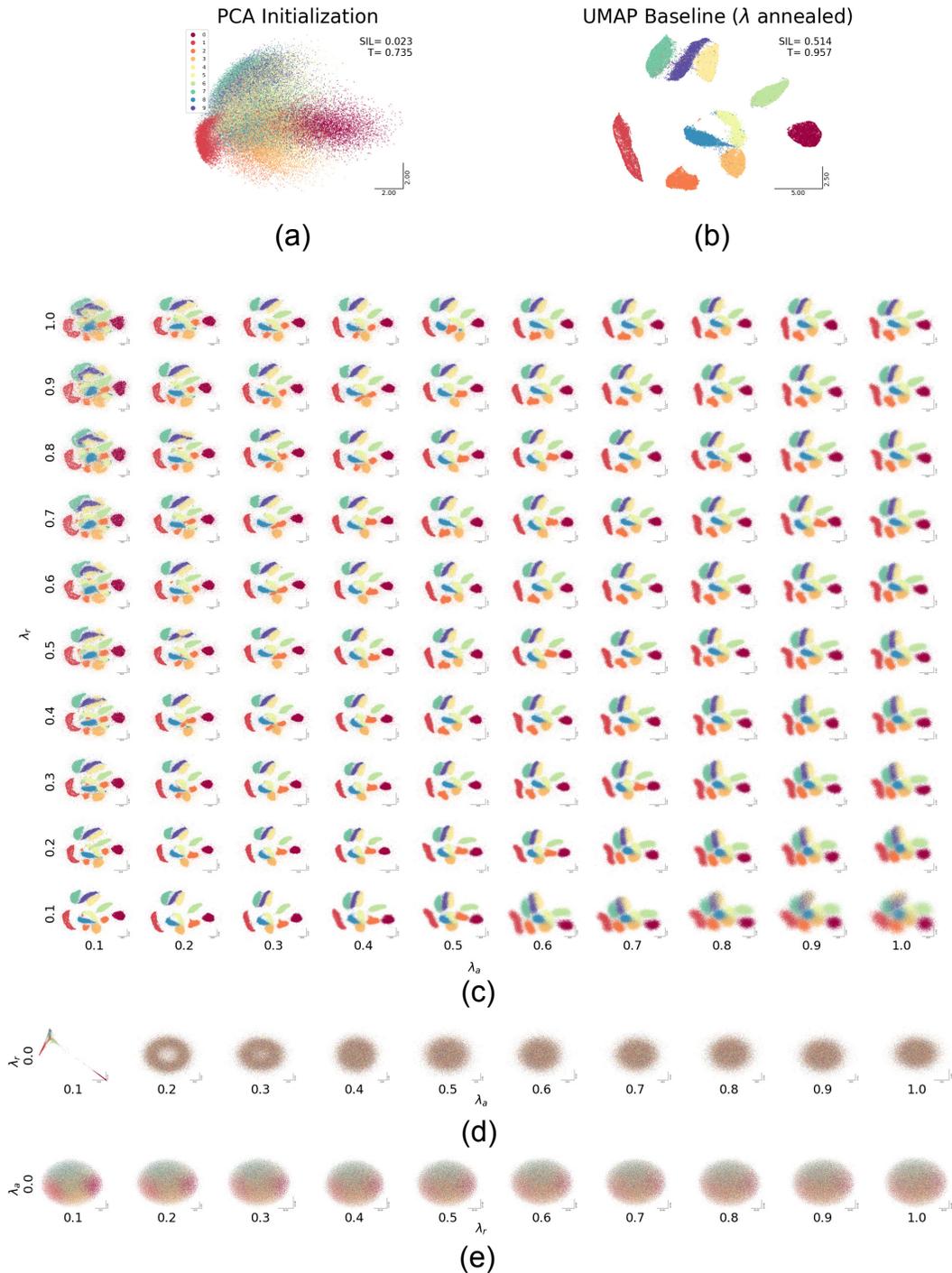


Figure 19: Varying  $\lambda_a$  and  $\lambda_r$  for the MNIST dataset using UMAP’s attraction and repulsion shapes. (a) Initialization for the embeddings. (b) Baseline when  $\lambda$  is annealed from 1. (c) The embeddings when  $\lambda_a$  and  $\lambda_r$  vary (without any annealing). (d) When  $\lambda_r$  is set to 0 (no repulsion), the attractive force alone cannot produce any cluster. (e) Similarly, when  $\lambda_a$  is set to 0 (no attraction), the repulsive force alone cannot produce any clusters.

1782  
 1783  
 1784  
 1785  
 1786  
 1787  
 1788  
 1789  
 1790  
 1791  
 1792  
 1793  
 1794  
 1795  
 1796  
 1797  
 1798  
 1799  
 1800  
 1801  
 1802  
 1803  
 1804  
 1805  
 1806  
 1807  
 1808  
 1809  
 1810  
 1811  
 1812  
 1813  
 1814  
 1815  
 1816  
 1817  
 1818  
 1819  
 1820  
 1821  
 1822  
 1823  
 1824  
 1825  
 1826  
 1827  
 1828  
 1829  
 1830  
 1831  
 1832  
 1833  
 1834  
 1835

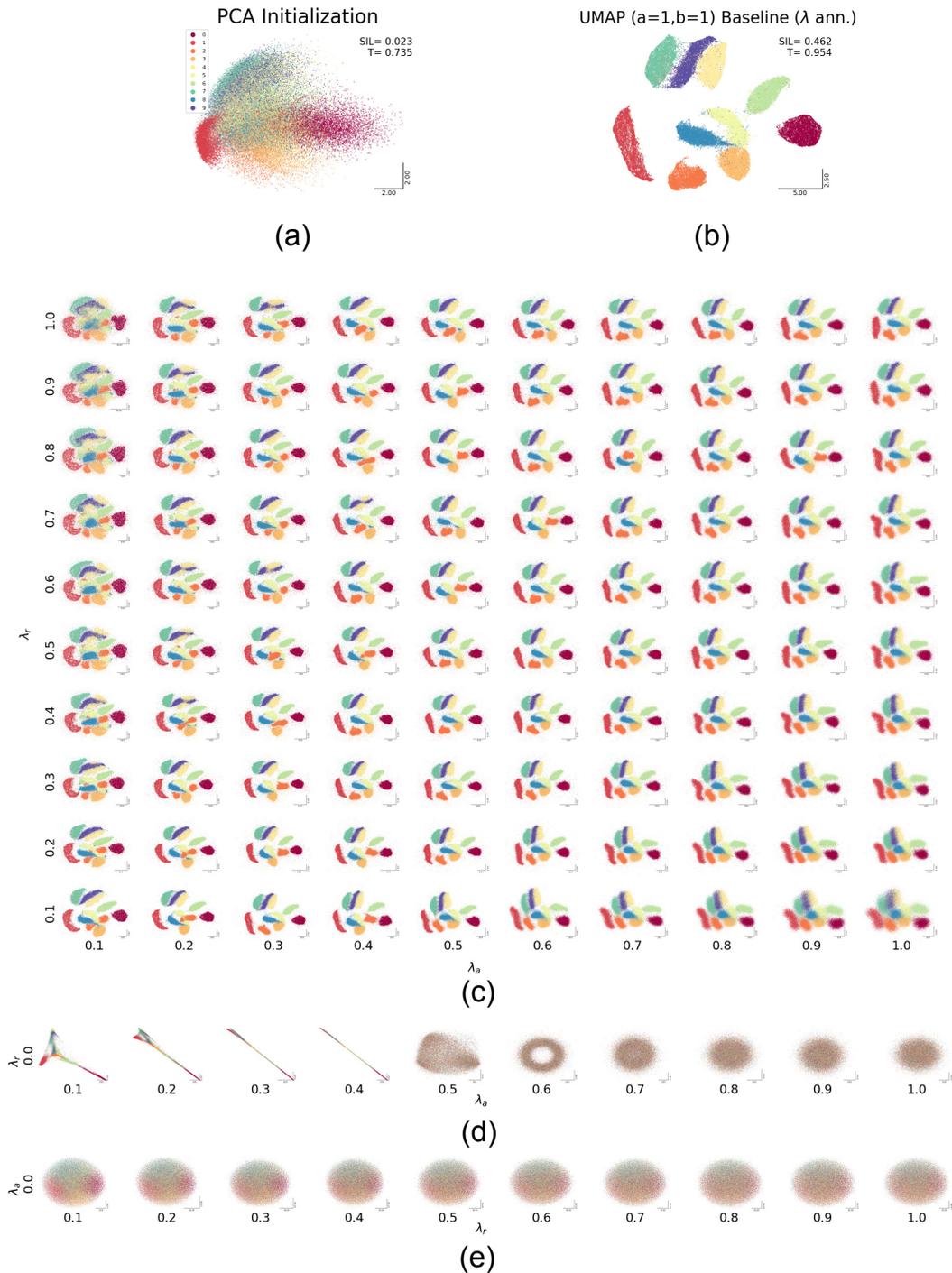


Figure 20: Varying  $\lambda_a$  and  $\lambda_r$  for the MNIST dataset using UMAP’s attraction and repulsion shapes (by setting  $a = 1$  and  $b = 1$ ). (a) Initialization for the embeddings. (b) Baseline when  $\lambda$  is annealed from 1. (c) The embeddings when  $\lambda_a$  and  $\lambda_r$  vary (without any annealing). (d) When  $\lambda_r$  is set to 0 (no repulsion), the attractive force alone cannot produce any cluster. (e) Similarly, when  $\lambda_a$  is set to 0 (no attraction), the repulsive force alone cannot produce any clusters.

1836  
 1837  
 1838  
 1839  
 1840  
 1841  
 1842  
 1843  
 1844  
 1845  
 1846  
 1847  
 1848  
 1849  
 1850  
 1851  
 1852  
 1853  
 1854  
 1855  
 1856  
 1857  
 1858  
 1859  
 1860  
 1861  
 1862  
 1863  
 1864  
 1865  
 1866  
 1867  
 1868  
 1869  
 1870  
 1871  
 1872  
 1873  
 1874  
 1875  
 1876  
 1877  
 1878  
 1879  
 1880  
 1881  
 1882  
 1883  
 1884  
 1885  
 1886  
 1887  
 1888  
 1889

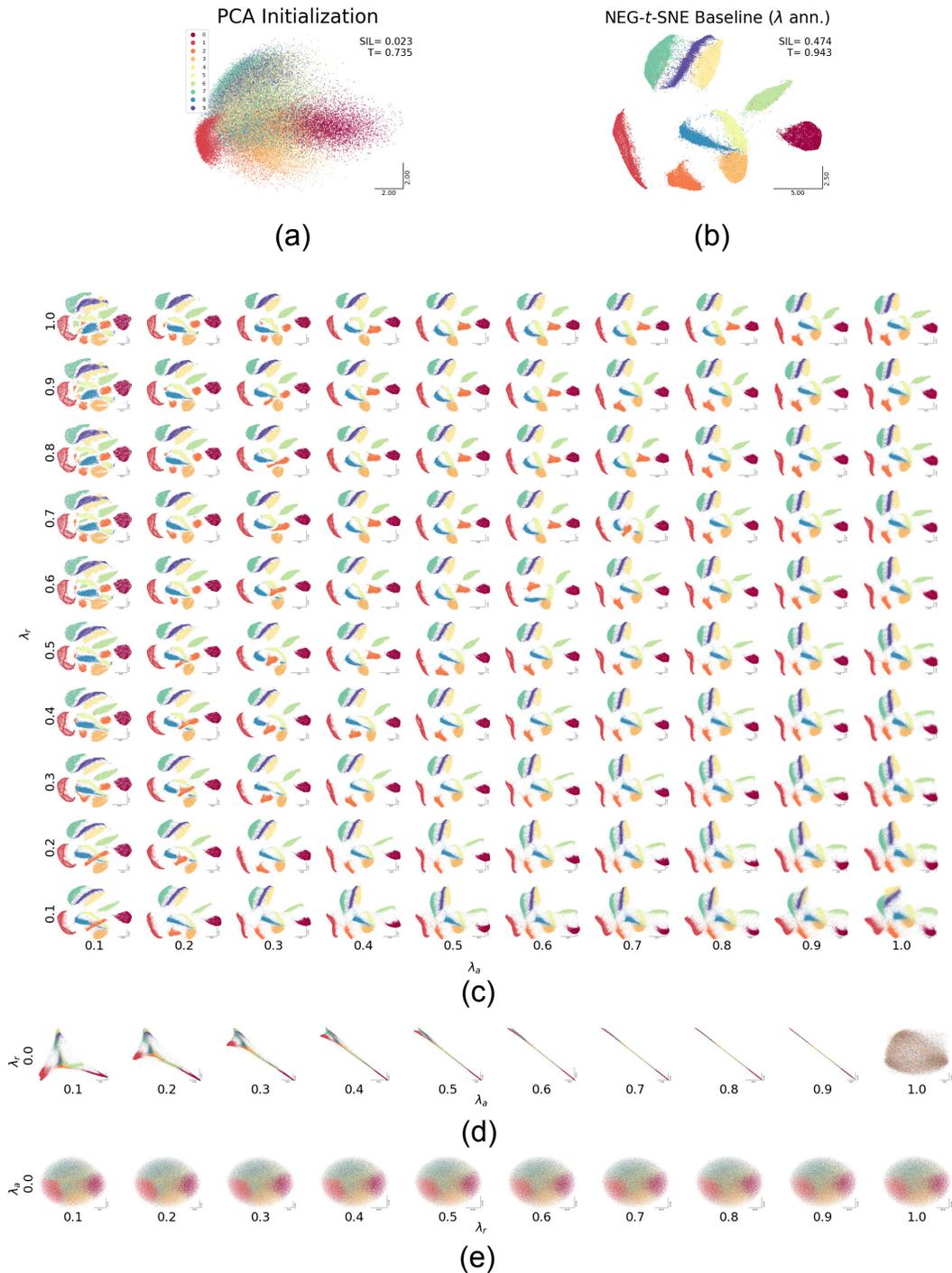


Figure 21: Varying  $\lambda_a$  and  $\lambda_r$  for the MNIST dataset using NEG-t-SNE’s attraction and repulsion shapes. (a) Initialization for the embeddings. (b) Baseline when  $\lambda$  is annealed from 1. (c) The embeddings when  $\lambda_a$  and  $\lambda_r$  vary (without any annealing). (d) When  $\lambda_r$  is set to 0 (no repulsion), the attractive force alone cannot produce any cluster. (e) Similarly, when  $\lambda_a$  is set to 0 (no attraction), the repulsive force alone cannot produce any clusters.

1890  
1891  
1892  
1893  
1894  
1895  
1896  
1897  
1898  
1899  
1900  
1901  
1902  
1903  
1904  
1905  
1906  
1907  
1908  
1909  
1910  
1911  
1912  
1913  
1914  
1915  
1916  
1917  
1918  
1919  
1920  
1921  
1922  
1923  
1924  
1925  
1926  
1927  
1928  
1929  
1930  
1931  
1932  
1933  
1934  
1935  
1936  
1937  
1938  
1939  
1940  
1941  
1942  
1943

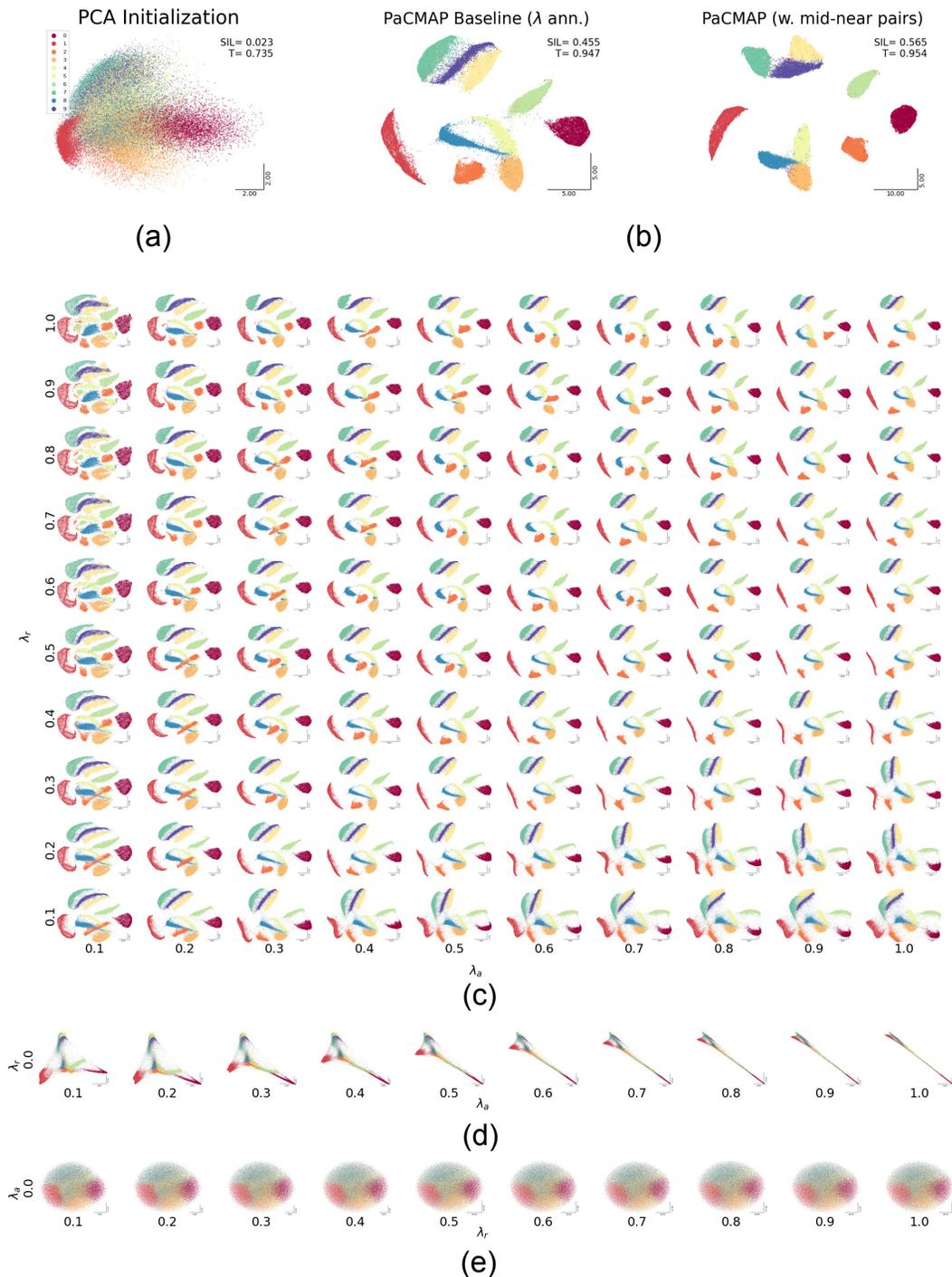


Figure 22: Varying  $\lambda_a$  and  $\lambda_r$  for the MNIST dataset using PaCMAP’s attraction and repulsion shapes. (a) Initialization for the embeddings. (b) Left: Baseline when  $\lambda$  is annealed from 1 (mid-near points are excluded to observe the interaction of attraction-repulsion alone), right: when mid-near points are considered. (c) The embeddings when  $\lambda_a$  and  $\lambda_r$  vary (without any annealing). (d) When  $\lambda_r$  is set to 0 (no repulsion), the attractive force alone cannot produce any cluster. (e) Similarly, when  $\lambda_a$  is set to 0 (no attraction), the repulsive force alone cannot produce any clusters.

1944  
 1945  
 1946  
 1947  
 1948  
 1949  
 1950  
 1951  
 1952  
 1953  
 1954  
 1955  
 1956  
 1957  
 1958  
 1959  
 1960  
 1961  
 1962  
 1963  
 1964  
 1965  
 1966  
 1967  
 1968  
 1969  
 1970  
 1971  
 1972  
 1973  
 1974  
 1975  
 1976  
 1977  
 1978  
 1979  
 1980  
 1981  
 1982  
 1983  
 1984  
 1985  
 1986  
 1987  
 1988  
 1989  
 1990  
 1991  
 1992  
 1993  
 1994  
 1995  
 1996  
 1997

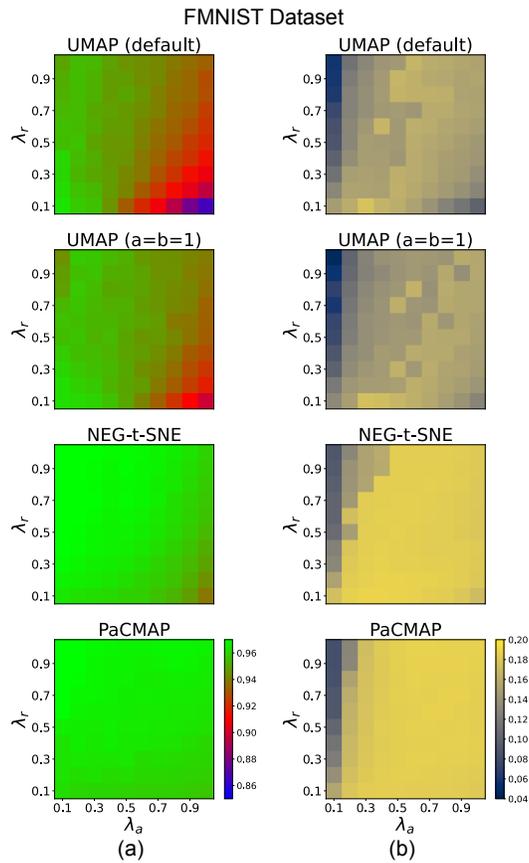


Figure 23: (a) Trustworthiness and (b) Silhouette score of different methods for the FMNIST dataset.

1998  
1999  
2000  
2001  
2002  
2003  
2004  
2005  
2006  
2007  
2008  
2009  
2010  
2011  
2012  
2013  
2014  
2015  
2016  
2017  
2018  
2019  
2020  
2021  
2022  
2023  
2024  
2025  
2026  
2027  
2028  
2029  
2030  
2031  
2032  
2033  
2034  
2035  
2036  
2037  
2038  
2039  
2040  
2041  
2042  
2043  
2044  
2045  
2046  
2047  
2048  
2049  
2050  
2051

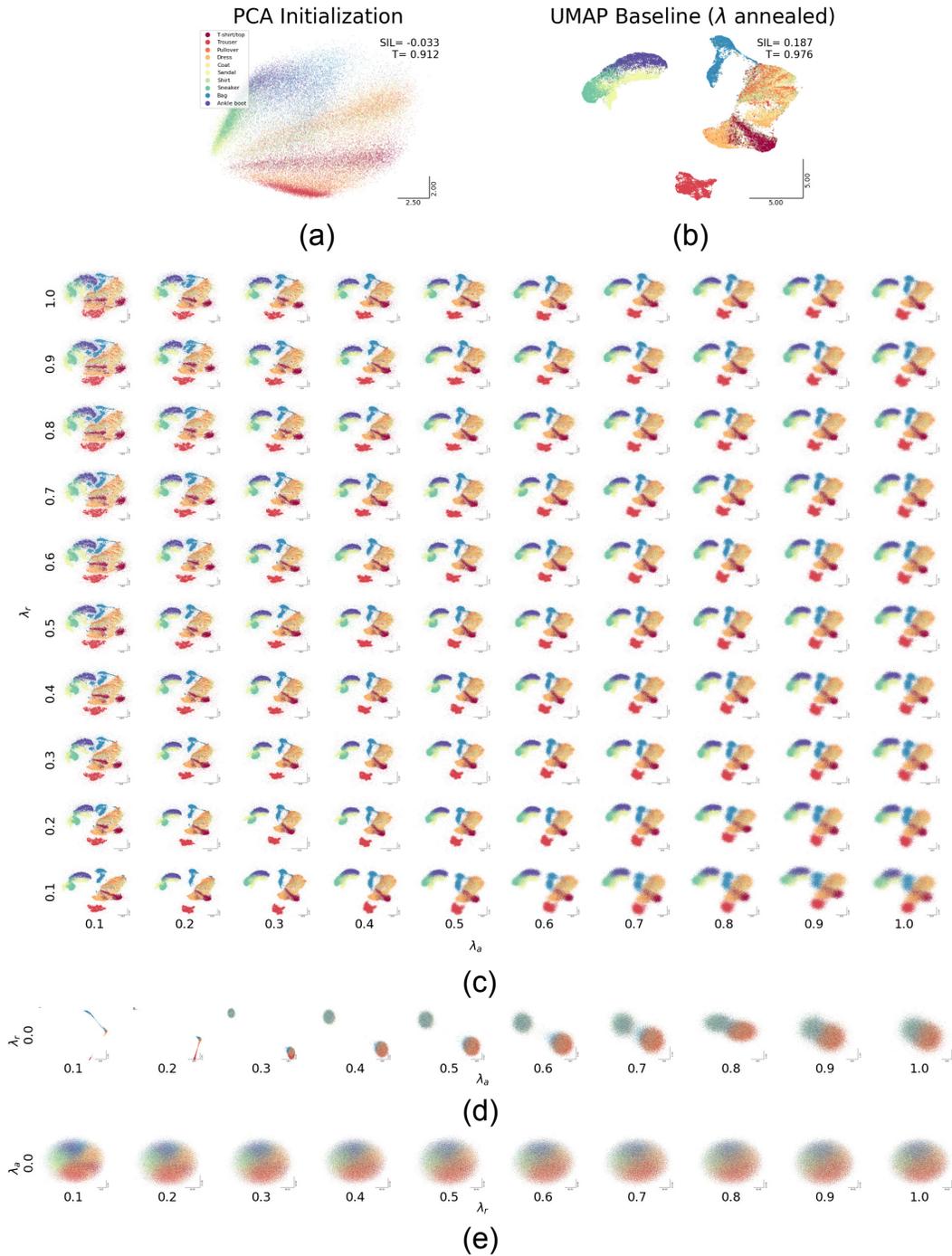


Figure 24: Varying  $\lambda_a$  and  $\lambda_r$  for the FMNIST dataset using UMAP’s attraction and repulsion shapes. (a) Initialization for the embeddings. (b) Baseline when  $\lambda$  is annealed from 1. (c) The embeddings when  $\lambda_a$  and  $\lambda_r$  vary (without any annealing). (d) When  $\lambda_r$  is set to 0 (no repulsion), the attractive force alone cannot produce any cluster. (e) Similarly, when  $\lambda_a$  is set to 0 (no attraction), the repulsive force alone cannot produce any clusters.

2052  
 2053  
 2054  
 2055  
 2056  
 2057  
 2058  
 2059  
 2060  
 2061  
 2062  
 2063  
 2064  
 2065  
 2066  
 2067  
 2068  
 2069  
 2070  
 2071  
 2072  
 2073  
 2074  
 2075  
 2076  
 2077  
 2078  
 2079  
 2080  
 2081  
 2082  
 2083  
 2084  
 2085  
 2086  
 2087  
 2088  
 2089  
 2090  
 2091  
 2092  
 2093  
 2094  
 2095  
 2096  
 2097  
 2098  
 2099  
 2100  
 2101  
 2102  
 2103  
 2104  
 2105

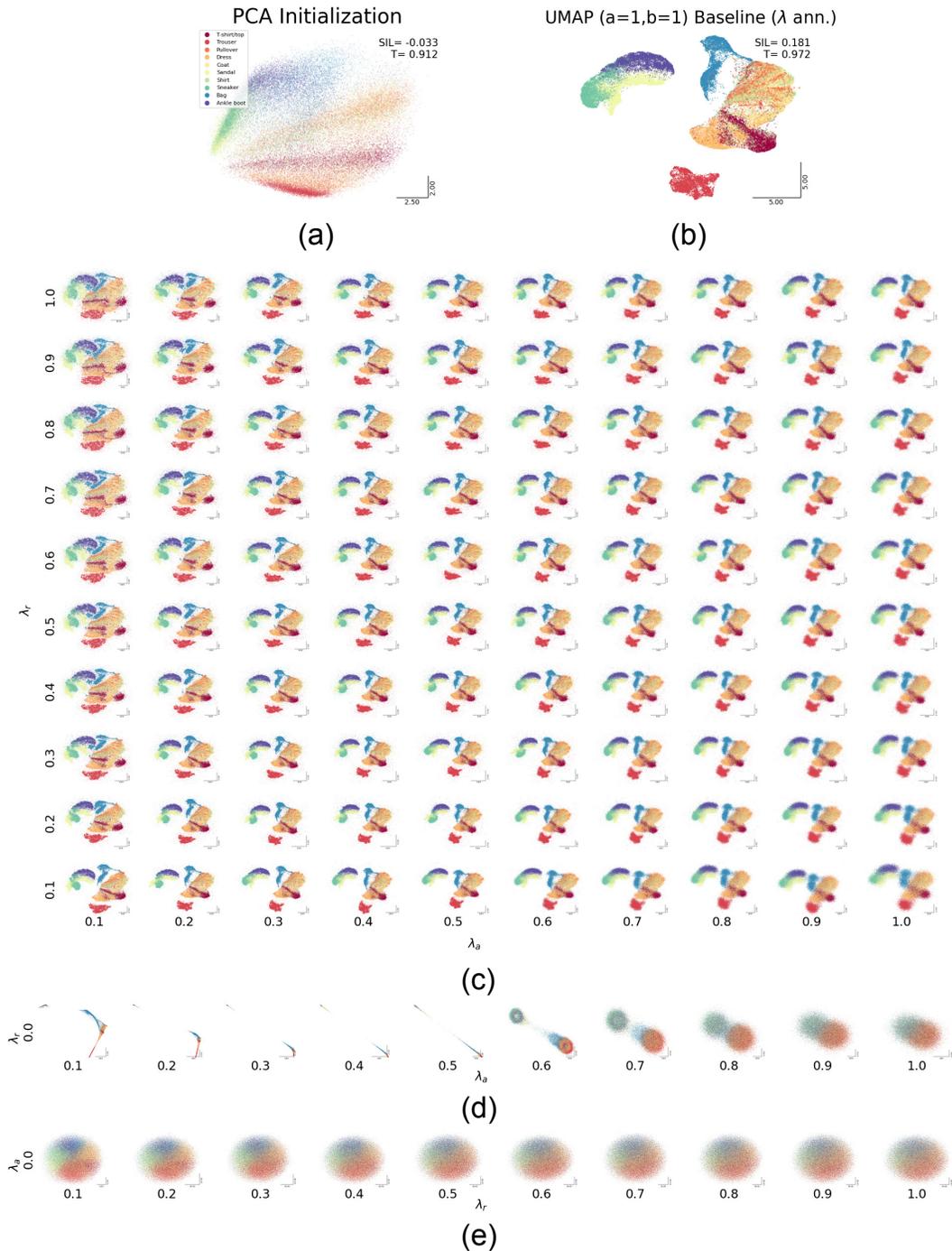


Figure 25: Varying  $\lambda_a$  and  $\lambda_r$  for the FMNIST dataset using UMAP’s attraction and repulsion shapes (by setting  $a = 1$  and  $b = 1$ ). (a) Initialization for the embeddings. (b) Baseline when  $\lambda$  is annealed from 1. (c) The embeddings when  $\lambda_a$  and  $\lambda_r$  vary (without any annealing). (d) When  $\lambda_r$  is set to 0 (no repulsion), the attractive force alone cannot produce any cluster. (e) Similarly, when  $\lambda_a$  is set to 0 (no attraction), the repulsive force alone cannot produce any clusters.

2106  
2107  
2108  
2109  
2110  
2111  
2112  
2113  
2114  
2115  
2116  
2117  
2118  
2119  
2120  
2121  
2122  
2123  
2124  
2125  
2126  
2127  
2128  
2129  
2130  
2131  
2132  
2133  
2134  
2135  
2136  
2137  
2138  
2139  
2140  
2141  
2142  
2143  
2144  
2145  
2146  
2147  
2148  
2149  
2150  
2151  
2152  
2153  
2154  
2155  
2156  
2157  
2158  
2159

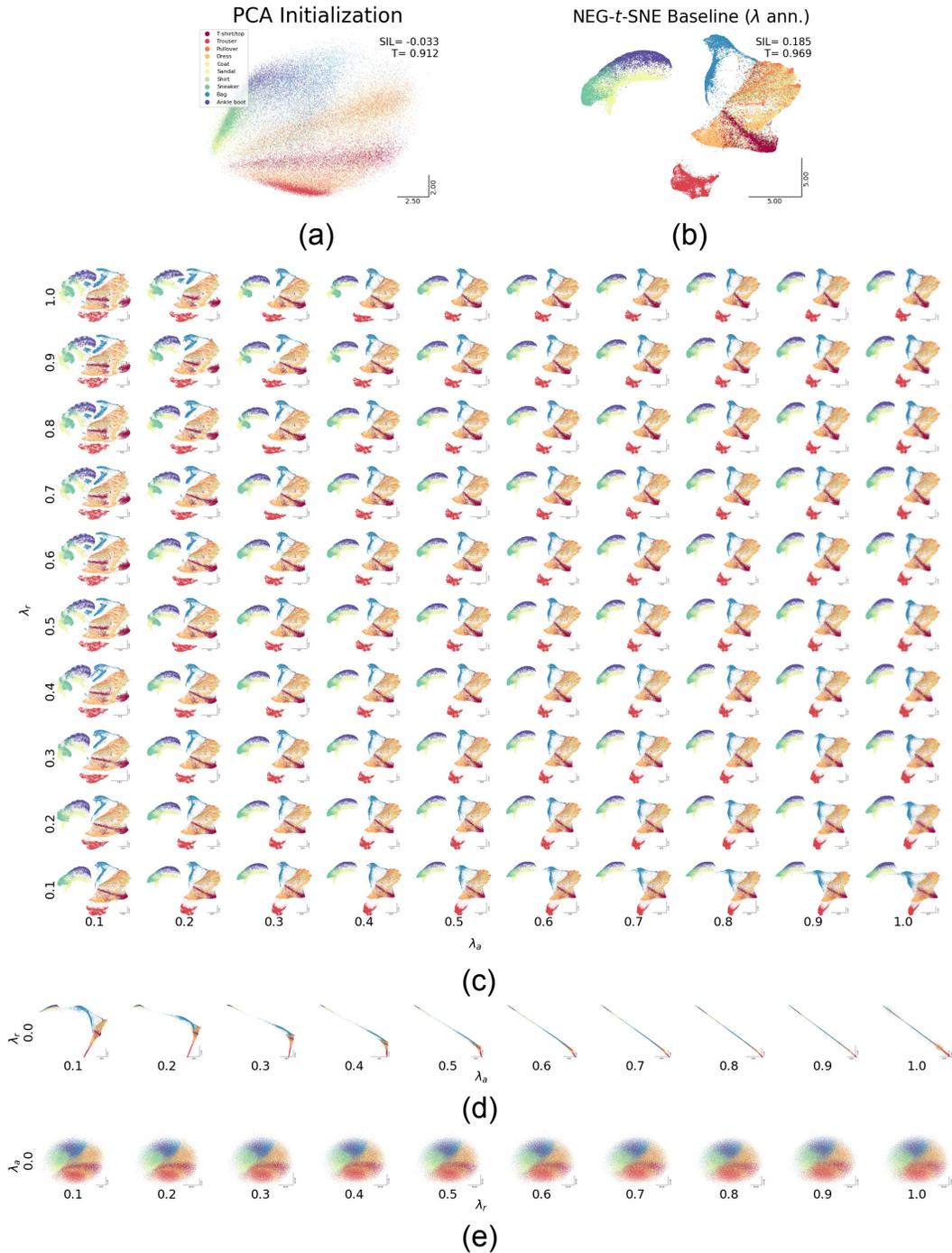
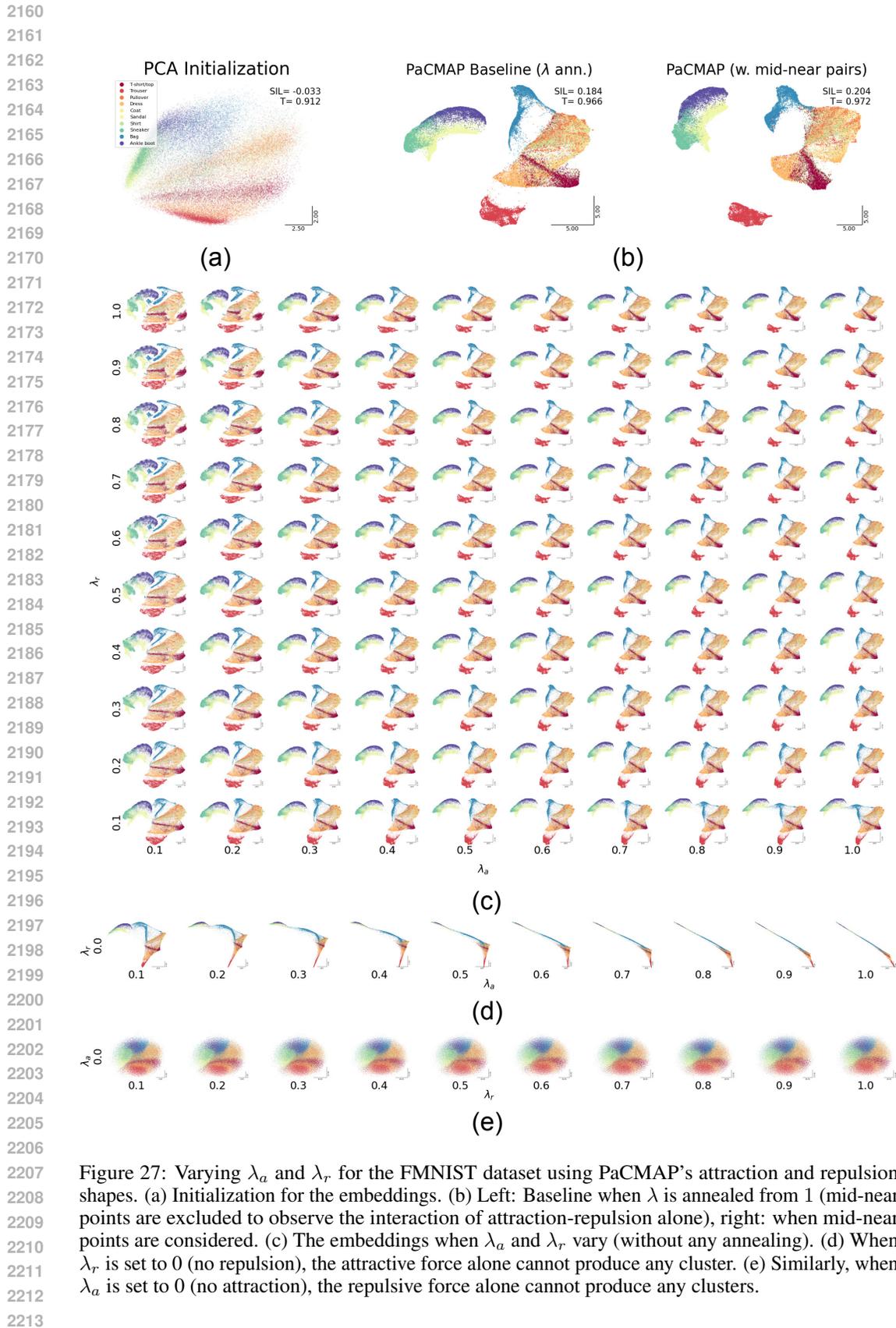


Figure 26: Varying  $\lambda_a$  and  $\lambda_r$  for the FMNIST dataset using NEG-t-SNE’s attraction and repulsion shapes. (a) Initialization for the embeddings. (b) Baseline when  $\lambda$  is annealed from 1. (c) The embeddings when  $\lambda_a$  and  $\lambda_r$  vary (without any annealing). (d) When  $\lambda_r$  is set to 0 (no repulsion), the attractive force alone cannot produce any cluster. (e) Similarly, when  $\lambda_a$  is set to 0 (no attraction), the repulsive force alone cannot produce any clusters.



2214  
 2215  
 2216  
 2217  
 2218  
 2219  
 2220  
 2221  
 2222  
 2223  
 2224  
 2225  
 2226  
 2227  
 2228  
 2229  
 2230  
 2231  
 2232  
 2233  
 2234  
 2235  
 2236  
 2237  
 2238  
 2239  
 2240  
 2241  
 2242  
 2243  
 2244  
 2245  
 2246  
 2247  
 2248  
 2249  
 2250  
 2251  
 2252  
 2253  
 2254  
 2255  
 2256  
 2257  
 2258  
 2259  
 2260  
 2261  
 2262  
 2263  
 2264  
 2265  
 2266  
 2267

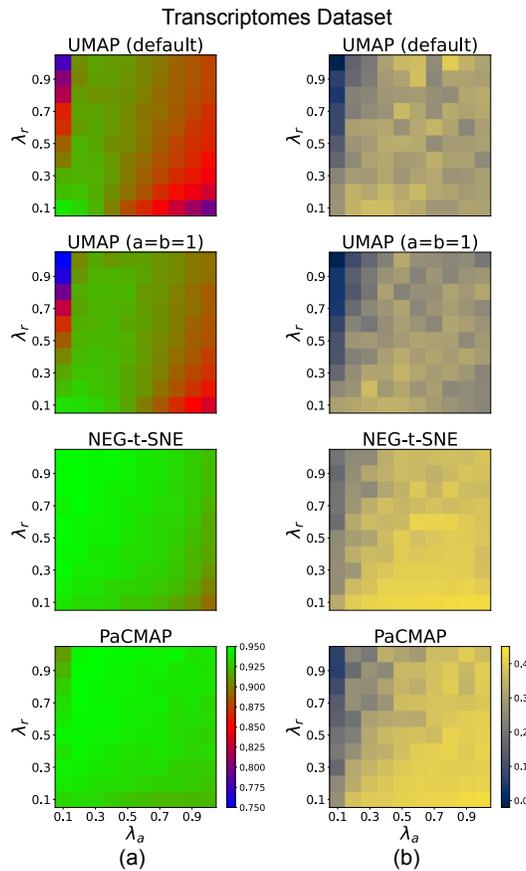


Figure 28: (a) Trustworthiness and (b) Silhouette score of different methods for the Single-cell transcriptomes dataset.

2268  
 2269  
 2270  
 2271  
 2272  
 2273  
 2274  
 2275  
 2276  
 2277  
 2278  
 2279  
 2280  
 2281  
 2282  
 2283  
 2284  
 2285  
 2286  
 2287  
 2288  
 2289  
 2290  
 2291  
 2292  
 2293  
 2294  
 2295  
 2296  
 2297  
 2298  
 2299  
 2300  
 2301  
 2302  
 2303  
 2304  
 2305  
 2306  
 2307  
 2308  
 2309  
 2310  
 2311  
 2312  
 2313  
 2314  
 2315  
 2316  
 2317  
 2318  
 2319  
 2320  
 2321

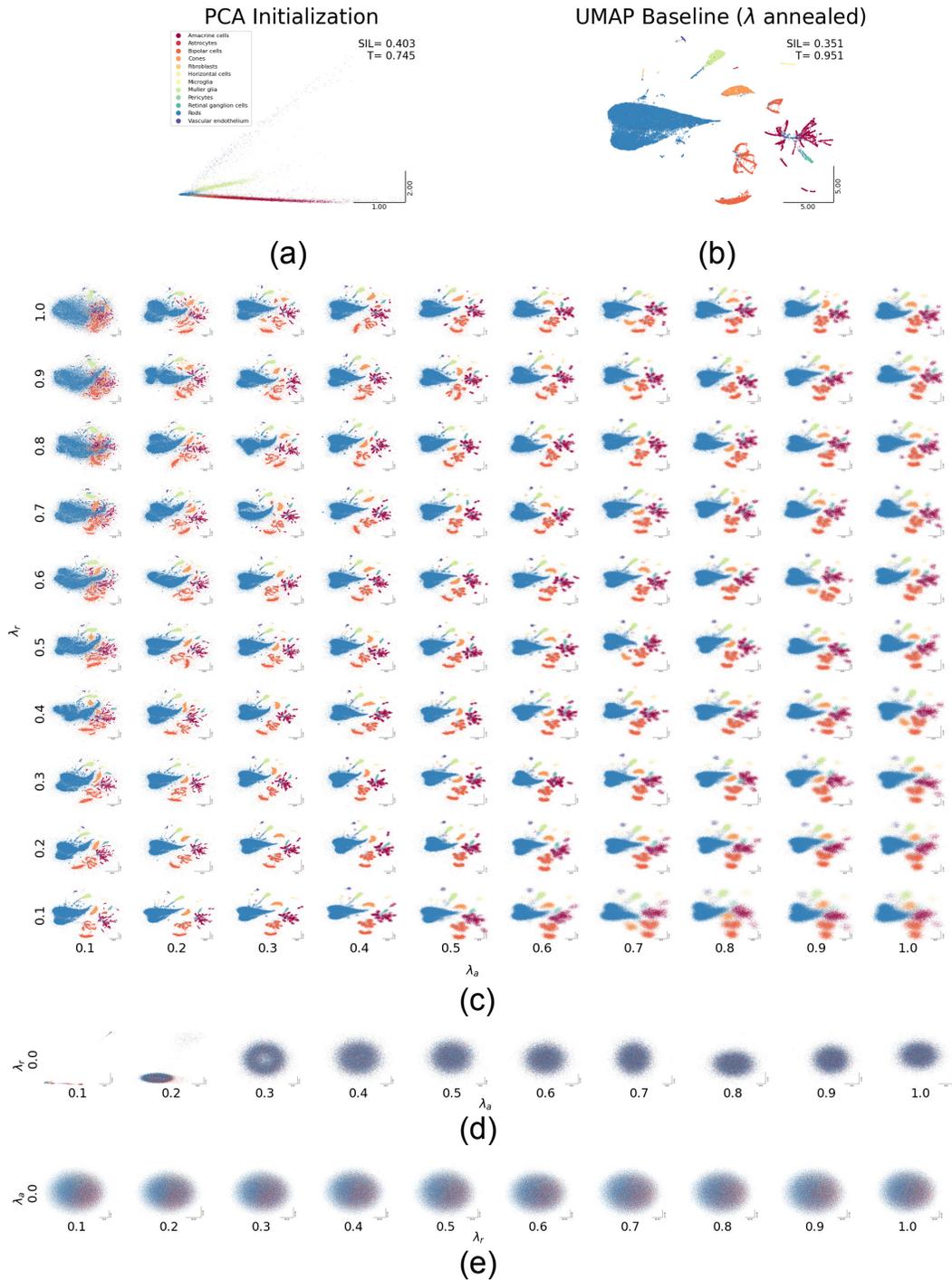


Figure 29: Varying  $\lambda_a$  and  $\lambda_r$  for the single-cell transcriptomes dataset using UMAP’s attraction and repulsion shapes. (a) Initialization for the embeddings. (b) Baseline when  $\lambda$  is annealed from 1. (c) The embeddings when  $\lambda_a$  and  $\lambda_r$  vary (without any annealing). (d) When  $\lambda_r$  is set to 0 (no repulsion), the attractive force alone cannot produce any cluster. (e) Similarly, when  $\lambda_a$  is set to 0 (no attraction), the repulsive force alone cannot produce any clusters.

2322  
 2323  
 2324  
 2325  
 2326  
 2327  
 2328  
 2329  
 2330  
 2331  
 2332  
 2333  
 2334  
 2335  
 2336  
 2337  
 2338  
 2339  
 2340  
 2341  
 2342  
 2343  
 2344  
 2345  
 2346  
 2347  
 2348  
 2349  
 2350  
 2351  
 2352  
 2353  
 2354  
 2355  
 2356  
 2357  
 2358  
 2359  
 2360  
 2361  
 2362  
 2363  
 2364  
 2365  
 2366  
 2367  
 2368  
 2369  
 2370  
 2371  
 2372  
 2373  
 2374  
 2375

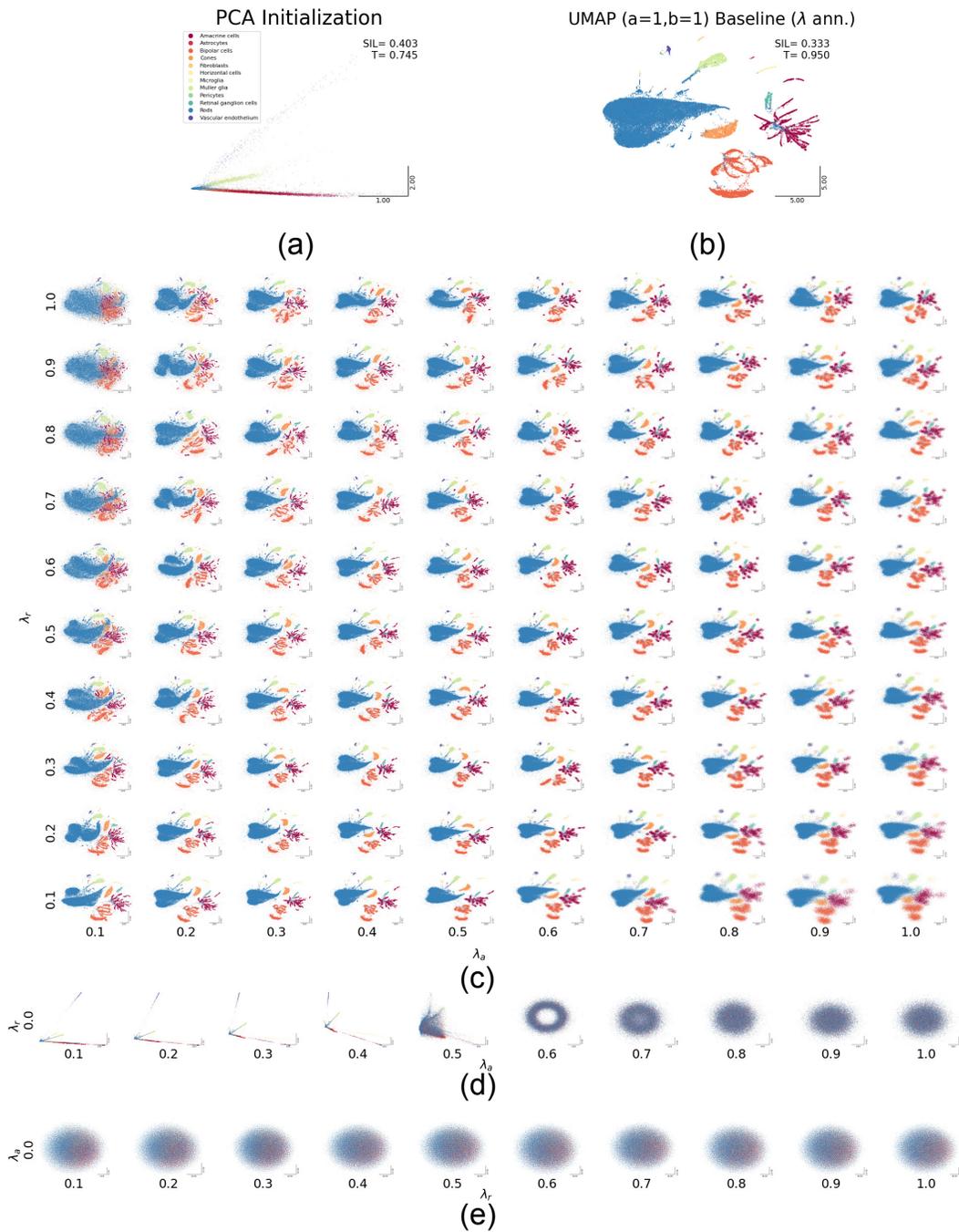


Figure 30: Varying  $\lambda_a$  and  $\lambda_r$  for the single-cell transcriptomes dataset using UMAP’s attraction and repulsion shapes (by setting  $a = 1$  and  $b = 1$ ). (a) Initialization for the embeddings. (b) Baseline when  $\lambda$  is annealed from 1. (c) The embeddings when  $\lambda_a$  and  $\lambda_r$  vary (without any annealing). (d) When  $\lambda_r$  is set to 0 (no repulsion), attractive force alone cannot produce any cluster. (e) Similarly, when  $\lambda_a$  is set to 0 (no attraction), repulsive force alone cannot produce any clusters.

2376  
2377  
2378  
2379  
2380  
2381  
2382  
2383  
2384  
2385  
2386  
2387  
2388  
2389  
2390  
2391  
2392  
2393  
2394  
2395  
2396  
2397  
2398  
2399  
2400  
2401  
2402  
2403  
2404  
2405  
2406  
2407  
2408  
2409  
2410  
2411  
2412  
2413  
2414  
2415  
2416  
2417  
2418  
2419  
2420  
2421  
2422  
2423  
2424  
2425  
2426  
2427  
2428  
2429

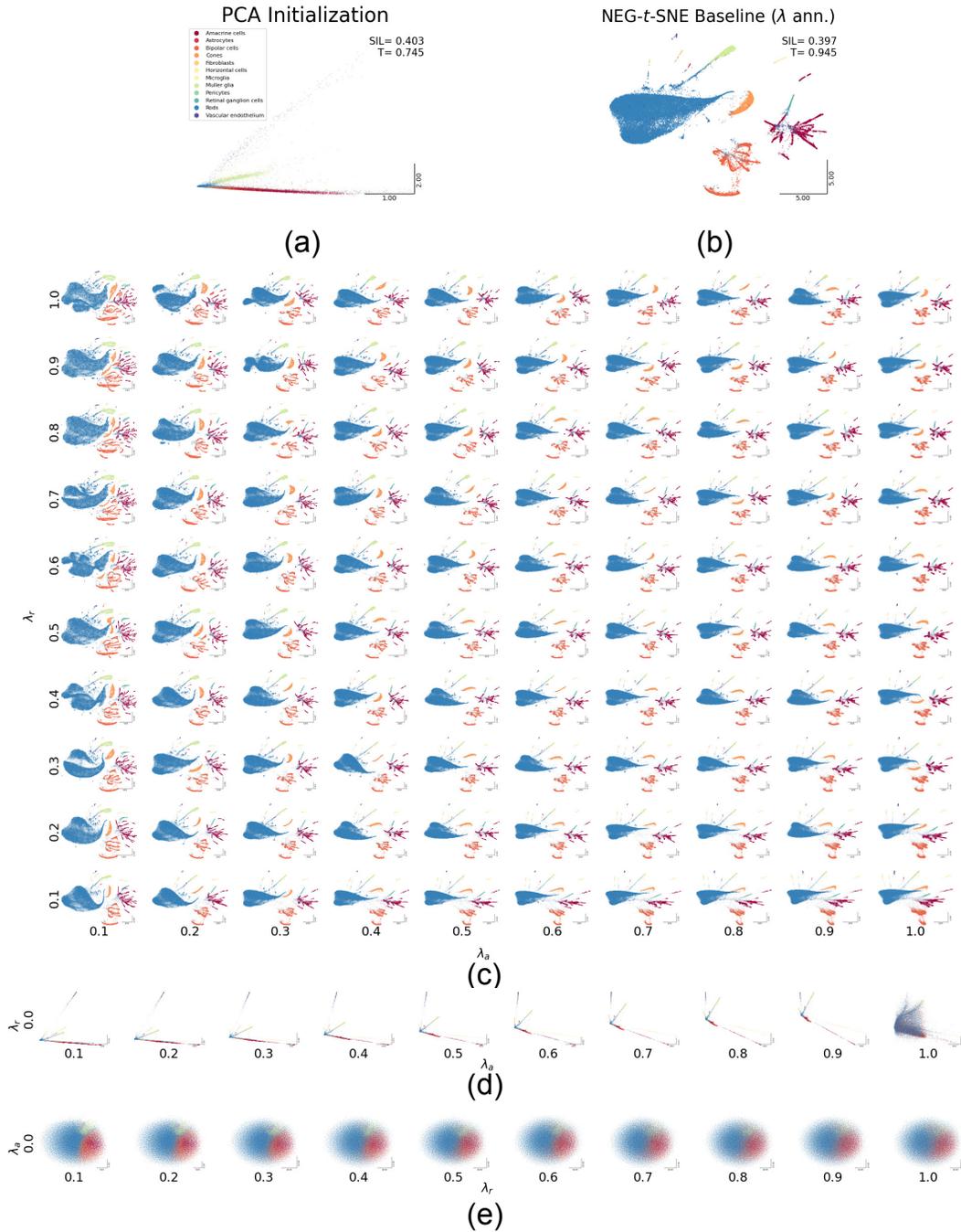


Figure 31: Varying  $\lambda_a$  and  $\lambda_r$  for the single-cell transcriptomes dataset using NEG-*t*-SNE’s attraction and repulsion shapes. (a) Initialization for the embeddings. (b) Baseline when  $\lambda$  is annealed from 1. (c) The embeddings when  $\lambda_a$  and  $\lambda_r$  vary (without any annealing). (d) When  $\lambda_r$  is set to 0 (no repulsion), the attractive force alone cannot produce distinct clusters. (e) Similarly, when  $\lambda_a$  is set to 0 (no attraction), the repulsive force alone cannot produce any clusters.

2430  
2431  
2432  
2433  
2434  
2435  
2436  
2437  
2438  
2439  
2440  
2441  
2442  
2443  
2444  
2445  
2446  
2447  
2448  
2449  
2450  
2451  
2452  
2453  
2454  
2455  
2456  
2457  
2458  
2459  
2460  
2461  
2462  
2463  
2464  
2465  
2466  
2467  
2468  
2469  
2470  
2471  
2472  
2473  
2474  
2475  
2476  
2477  
2478  
2479  
2480  
2481  
2482  
2483

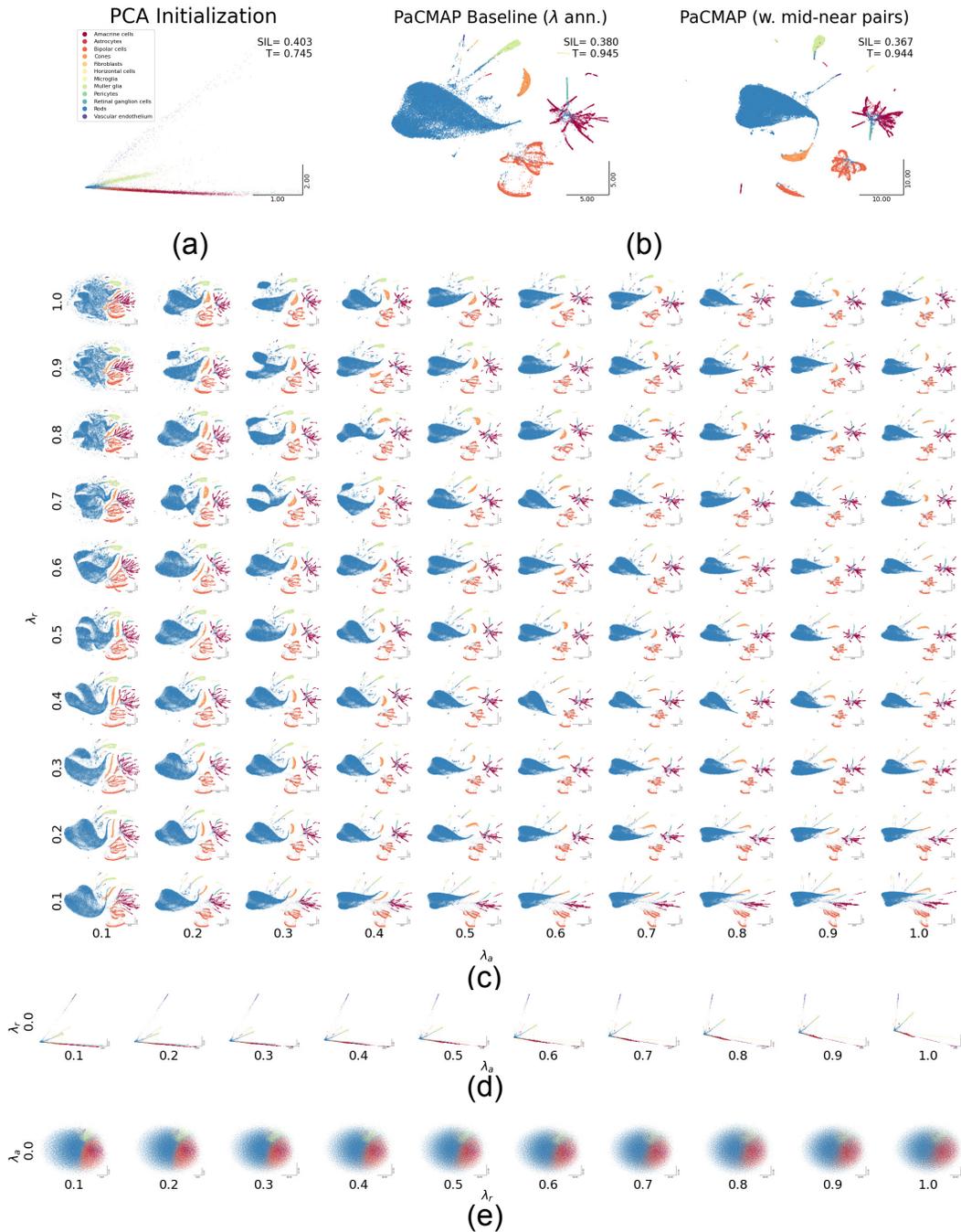


Figure 32: Varying  $\lambda_a$  and  $\lambda_r$  for the single-cell transcriptomes dataset using PaCMAP’s attraction and repulsion shapes. (a) Initialization for the embeddings. (b) Left: Baseline when  $\lambda$  is annealed from 1 (mid-near points are excluded to observe the interaction of attraction-repulsion alone), right: when mid-near points are considered. (c) The embeddings when  $\lambda_a$  and  $\lambda_r$  vary (without any annealing). (d) When  $\lambda_r$  is set to 0 (no repulsion), the attractive force alone cannot produce any cluster. (e) Similarly, when  $\lambda_a$  is set to 0 (no attraction), the repulsive force alone cannot produce any clusters.

## J MIX AND MATCH APPROACH

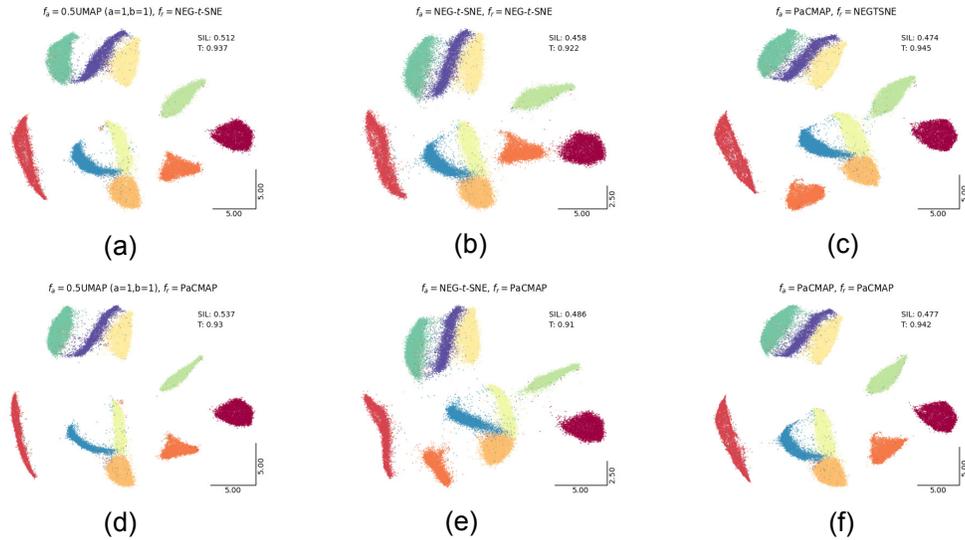


Figure 33: Embedding of MNIST by mixing and matching attraction and repulsion shapes with a constant learning rate ( $\lambda = 1$ ). We used attraction shapes that follow Proposition 4.1 and repulsion shapes that are finite. Top row: UMAP, NEG-*t*-SNE, and PaCMAP’s attraction shapes with NEG-*t*-SNE’s repulsion shape. Bottom row: UMAP, NEG-*t*-SNE, and PaCMAP’s attraction shapes with PaCMAP’s repulsion shape.

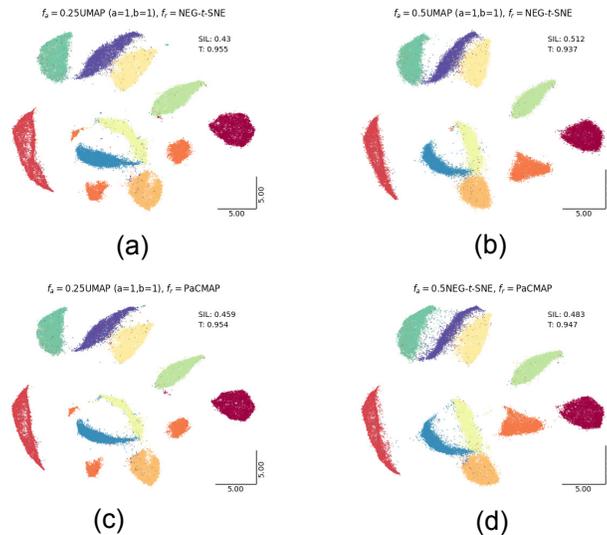


Figure 34: Embedding of MNIST by mixing and matching attraction and repulsion shapes with a constant learning rate ( $\lambda = 1$ ). We kept the attraction shape confined to  $[-0.5, 0]$  to prevent any flips during the attractive update. Top row: UMAP and NEG-*t*-SNE’s attraction shapes with NEG-*t*-SNE’s repulsion shape. Bottom row: UMAP and NEG-*t*-SNE’s attraction shapes with PaCMAP’s repulsion shape.

In our analysis, we considered the attraction and the repulsion shapes to be independent of each other. It thus suggests that we can combine attraction and repulsion shapes derived from different loss functions and principles. Figure 33 shows a few such examples. For this, we restricted the attraction shapes to  $[-1, 0]$  (to work with Proposition 4.1) so that they become less sensitive to learning rates.

We considered finite repulsion shapes, hence mitigating the effect of large repulsive forces for smaller distances. We held the learning rate constant ( $\lambda = 1$ ) and used the same optimization (of UMAP) for all of them, so that any changes arise from the shapes rather than learning-rate scheduling or optimization scheme. Overall, UMAP’s attraction shape (by  $f_a = 0.5f_a^U$ ) makes better clusters (higher silhouette score), while PaCMAP’s attraction shape provides better local structure (higher trustworthiness). The latter is primarily due to PaCMAP’s attraction shape being within  $[-0.5, 0]$ , which prevents any flips during the updates. By confining UMAP and NEG- $t$ -SNE’s attraction shape within  $[-0.5, 0]$  (by  $f_a = 0.25f_a^U$  and  $f_a = 0.5f_a^N$ , respectively), we can improve the local structure at the expense of clustering performance (Fig. 34).

## K IMPLEMENTATION DETAILS

For analysis, we implemented our own UMAP algorithm. We used `numba` (Lam et al., 2015) to compute an exact nearest neighbor graph (instead of an approximate one) with  $k = 15$  and `scikit-learn`’s (Pedregosa et al., 2011) implementation of the PCA algorithm for PCA initialization. We used this to produce and quantify the embeddings in Figs. 1, 2, 3, 4, 6, 8, 9, and 10. We also used the same implementation when we changed the attraction and the repulsion shapes to those of the alternative methods (Figs. 18- 34, unless otherwise stated). The trustworthiness and silhouette scores were computed using the corresponding function from the `scikit-learn` package.

The mappings shown in Figs. 2, 3, 9, 10, and 15 are rotated to a reference embedding ((a) for each respective Figures). To achieve this, we performed Procrustes alignment of the embeddings by normalizing them (zero mean and unit norm) and then using SciPy’s (Virtanen et al., 2020) `orthogonal_procrustes` method to extract rotation and scaling parameters.

To compare with Neg- $t$ -SNE in Fig. 11, we used the original implementation of the contrastive embedding framework for both the UMAP and the Neg- $t$ -SNE algorithms (available at <https://github.com/berenslab/contrastive-ne>).

PaCMAP and LocalMAP embeddings in Figs. 15, 16, 22 (b) (right), 27 (b) (right), and 32 (b) (right) were obtained using the official PaCMAP package (available at <https://github.com/YingfanWang/PaCMAP>).

The codes for displaying the embeddings of the PubMed dataset in Fig. 7 are adapted from <https://github.com/berenslab/pubmed-landscape>.