Clustering and visualization of single-cell RNA-seq data using path metrics

Andriana Manousidaki[1,✉], **Anna Little**[2,✉], **and Yuying Xie**[3,✉]

[1]Department of Statistics and Probability, Michigan State University, East Lansing, 48824, United States of America
[2]Department of Mathematics, University of Utah, Salt Lake City, 84112, United States of America.
[3]Department of Computational Mathematics, Science and Engineering, Michigan State University, East Lansing, 48824, United States of America.

Recent advances in single-cell technologies have enabled high-resolution characterization of tissue and cancer compositions. Although numerous tools for dimension reduction and clustering are available for single-cell data analyses, these methods often fail to simultaneously preserve local cluster structure and global data geometry. This article explores the application of power-weighted path metrics for the analysis of single cell RNA data.

Extensive experiments on single cell RNA sequencing data sets confirm the usefulness of path metrics for dimension reduction and clustering. Distances between cells are measured in a data-driven way which is both density sensitive (decreasing distances across high density regions) and respects the underlying data geometry. By combining path metrics with multidimensional scaling, a low dimensional embedding of the data is obtained which respects both the global geometry of the data and preserves cluster structure. We evaluate the method both for clustering quality and geometric fidelity, and it outperforms other algorithms on a wide range of bench marking data sets.

scRNAseq | clustering | path metrics

Correspondence: ***little@math.utah.edu and xyy@msu.edu***

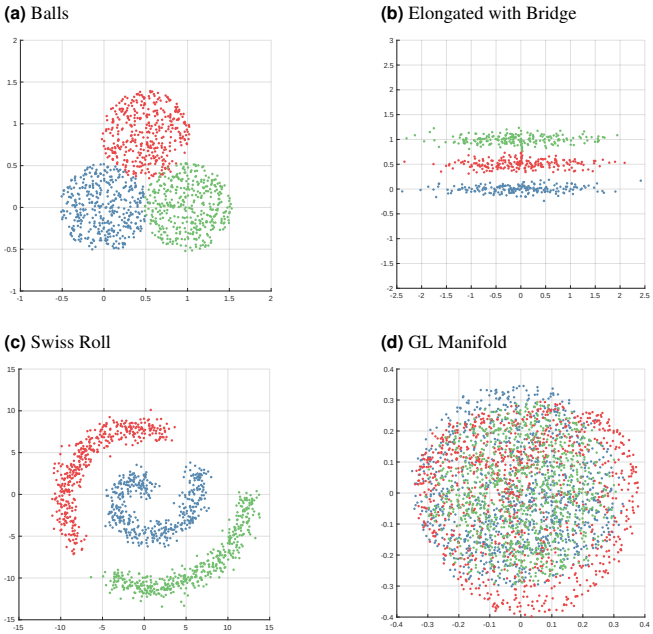

Fig. 1. Toy Data Sets. 1a and 1b show the 2-dimensional data sets. 1c plots the first two coordinates of the Swiss roll. 1d shows the 2-dimensional PCA plot of the SO(3) manifolds.

Introduction

The advance in single-cell RNA-seq (scRNA-seq) technologies in recent years has enabled the simultaneous measurement of gene expression at the single-cell level (1–3). This opens up new possibilities to detect previously unknown cell populations, study cellular development and dynamics, and characterize cell composition within bulk tissues. Despite its similarity with bulk RNAseq data, scRNAseq data tends to have larger variation and larger amount of missing values (dropouts) due to the low amount of initial mRNA per cell. To address these challenges, numerous computational algorithms have been proposed focusing on different aspects. Given a collection of single cell transcriptomes from scRNAseq, one of the most powerful applications is to identify and characterize subpopulations, e.g., cell types or cell states. Numerous clustering approaches have been developed such as k-means based methods SC3 (4), SIMLR, and RaceID (5); hierarchical clustering based methods CIDR (6), BackSPIN (7), and pcaReduce (8); and graph based methods Rphenograph (9), SNN-Cliq (10), SSNN-Louvain (11), Seurat (12), and scanpy (13). In order to visualize and characterize relationships between cell types, it is important not just to cluster the data, but also to represent it in a low-dimensional space. Many low-dimensional embedding methods have been proposed including UMAP (14), t-SNE (15), PHATE (16), and LargeVis (17). However a key challenge for embedding methods is to simultaneously reduce cluster variance and preserve the global geometry, including the distances between clusters and cluster shapes.

To achieve both of these goals, we propose an embedding method based on the *power weighted path metric*. These metrics balance density and geometry considerations in the data, making them useful for many machine learning tasks such as clustering and semi-supervised learning (18–26). They have performed well in numerous applications especially imaging (24, 25, 27, 28), but their usefulness for the analysis of single cell RNA sequence data remains unexplored. Because these metrics are density-sensitive, they reduce cluster variance; however these metrics also capture global distance information, and thus preserve global geometry. Using the path metric embedding to cluster the data thus yields a clustering method which balances density-based and geometric information.

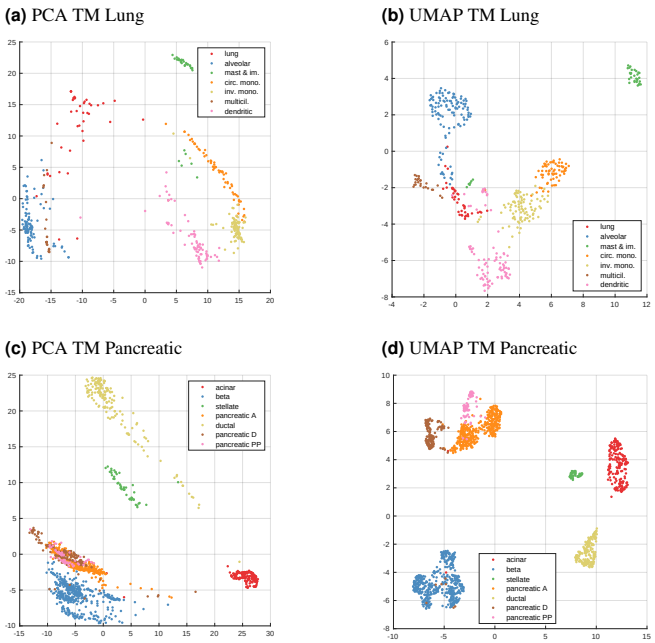

Fig. 2. Tabula Muris Data Sets. For TM lung, 'lung' includes neuroendocrine cells and unknown cells; 'alveolar' includes alveolar ep 1&2, club and basal; 'mast & im.' includes mast cells and unknown immune cells; 'circ. mono.' is circulating monocytes; 'inv. mono.' is invading monocytes; 'multicil.' is multiciliated; 'dendritic' includes dendritic, alveolar macro, and interstital macro.

Materials and Methods

We first introduce our theoretical framework in Section ; Section then describes the details of the proposed algorithm, and Section describes metrics for assessment.

Path Metrics. We first define a family of power weighted path metrics parametrized by $1 \leq p < \infty$.
Definition 1: Given a discrete data set X, the discrete p-power weighted path metric between $a, b \in X$ is defined as

$$\ell_p(a,b) := \inf_{(x_0,\ldots,x_s)} \left(\sum_{i=0}^{s-1} \|x_{i+1} - x_i\|_2^p \right)^{\frac{1}{p}},$$

where the infimum is taken over all sequences of points $x_0, \ldots, x_s$ in X with $x_0 = a$ and $x_s = b$.
Note as $p \to \infty$, ℓ_p converges to the "bottleneck edge" distance

$$\ell_\infty(a,b) := \inf_{(x_0,\ldots,x_s)} \max_i \|x_{i+1} - x_i\|_2,$$

which is well studied in the computer science literature (29–32). Two points are close in ℓ_∞ if they are connected by a high-density path through the data, regardless of how far apart the points are. On the other hand, when $p = 1$, ℓ_1 reduces to Euclidean distance. If path edges are furthermore restricted to lie in a nearest neighbor graph, ℓ_1 approximates the geodesic distance between the points, i.e. the length of the shortest path lying on the underlying data structure. The parameter p governs a trade-off between these two extremes, i.e. it determines how to balance density and geometry considerations when determining which data points should be considered close.

The relationship between ℓ_p and density can be made precise. Assume n independent samples from a continuous, nonzero density function f supported on a d-dimensional, compact Riemannian manifold $\mathcal{M}$ (a manifold is a smooth, locally linear surface; see (33)). Then for $p > 1$, $\ell_p(a,b)$ converges (after appropriate normalization) to

$$\mathcal{L}_p(a,b) := \inf_\gamma \left(\int f(\gamma(t))^{-\frac{(p-1)}{d}} |\gamma'(t)| \, dt \right)^{\frac{1}{p}}, \quad \textbf{(1)}$$

as $n \to \infty$, where the infimum is taken over all smooth curves $\gamma : [0,1] \to \mathcal{M}$ connecting a, b. Note $|\gamma'(t)|$ is simply the arclength element on $\mathcal{M}$, so $\mathcal{L}_1$ reduces to the standard geodesic distance. When $p \neq 1$, one obtains a density-weighted geodesic distance. The optimal $\mathcal{L}_p$ path is not necessarily the most direct: a detour may be worth it if it allows the path to stay in a high-density region; see Figure 3. Thus the metric is *density-sensitive*, in that distances across high-density regions are smaller than distances across low-density regions, a desirable property for many machine learning tasks (34), including trajectory estimation for developmental cells and cancer cells. However the metric is also *geometry preserving*, since it is computed by path integrals on $\mathcal{M}$. The parameter p controls the balance of these two properties: when p is small, $\mathcal{L}_p$ depends mainly on the geometry of the data, while for large p, $\mathcal{L}_p$ is primarily determined by data density. Although path metrics are defined in a complete graph, i.e. Definition 1 considers *every* path in the data connecting a, b, recent work (24, 35–37) has established that it is sufficient to only consider paths in a K-nearest neighbors (KNN) graph, as long as $K \geq C \log n$ for a constant C depending on p, d, f, and the geometry of the data. By restricting to a KNN graph, all pairwise path distances can be computed in $O(Kn^2)$ with Dijkstra's algorithm.

Algorithm. We consider a noisy data set of n data points $\widetilde{x}_1, \ldots, \widetilde{x}_n \in \mathbb{R}^d$, which form the rows of noisy data matrix $\widetilde{X} \in \mathbb{R}^{n \times d}$. We first denoise the data with a local averaging procedure, which has been shown to be advantageous for manifold plus noise data models (38). More specifically, we replace $\widetilde{x}_i$ with its local average:

$$x_i := \frac{1}{K_1} \sum_{j \in \mathcal{N}_{i,K_1}} \widetilde{x}_j \quad , \quad \mathcal{N}_{i,K_1} = \{ j : \widetilde{x}_j \text{ is a } K_1\text{NN of } \widetilde{x}_i \},$$

and let $X \in \mathbb{R}^{n \times d}$ denote the denoised data matrix.

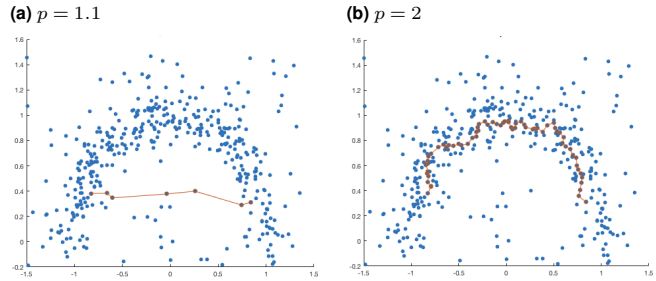

Fig. 3. Optimal ℓ_p path between two points in a moon data set.

Algorithm 1 PM-MDS

1: **Input:** noisy data $\widetilde{X} \in \mathbb{R}^{n\times d}$, parameter p, number of clusters k
2: **Optional input:** $K_1, K_2, r_{\min}, r_{\max}, \tau$
3: (Defaults: $12, n \wedge 500$, 3, 39, 0.01)
4: **Output:** PM-MDS embedding $Y \in \mathbb{R}^{n\times r}$, label vector $\hat{\ell} \in [k]^n$
5:
6: % Denoise data:
7: $x_i \leftarrow \frac{1}{K_1}\sum_{j\in\mathcal{N}_{i,K_1}} \widetilde{x}_i$
8:
9: % Compute path metrics:
10: $\mathcal{G}^p_{K_2} \leftarrow K_2$NN graph on X with edge weights $\|x_i - x_j\|^p$
11: $D^p_{ij} \leftarrow$ length of shortest path connecting x_i, x_j in $\mathcal{G}^p_{K_2}$
12: $(D_{\text{PM}})_{ij} \leftarrow (D^p_{ij})^{\frac{1}{p}}$
13:
14: % Compute PM-MDS embedding:
15: $B \leftarrow -\frac{1}{2} J D^{(2)}_{\text{PM}} J$
16: $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_n) \leftarrow$ eigenvalues of B in descending order
17: $V = (v_1, \ldots, v_n) \leftarrow$ corresponding eigenvectors of B
18: $r \leftarrow$ index maximizing λ_i/λ_{i+1} for i satisfying $r_{\min} \leq i \leq r_{\max}$, $\lambda_i/\lambda_1 \geq \tau$
19: $Y \leftarrow (\sqrt{\lambda_1}v_1, \ldots, \sqrt{\lambda_r}v_r) \in \mathbb{R}^{n\times r}$
20:
21: % Cluster the data:
22: $\hat{\ell} \leftarrow$ constrained k-means(Y, k)

We then fix p and compute the p-power weighted path distance between all points in X to obtain pairwise distance matrix $D_{\text{PM}} \in \mathbb{R}^{n\times n}$. More precisely, we let $\mathcal{G}^p_{K_2} = (X, E)$ be the graph on X where x_i, x_j are connected with edge weight $E_{ij} = \|x_i - x_j\|^p$ if x_i is a K_2NN of x_j or x_j is a K_2NN of x_i. We then compute D^p_{ij} as the length of the shortest path connecting x_i, x_j in $\mathcal{G}^p_{K_2}$, and define D_{PM} by $(D_{\text{PM}})_{ij} = (D^p_{ij})^{\frac{1}{p}}$.

We next apply classical multidimensional scaling (39) to obtain a low-dimensional embedding which preserves the path metrics. Specifically, we define the PM-MDS matrix $B = -\frac{1}{2} J D^{(2)}_{\text{PM}} J$ where $J = I_n - \frac{1}{n}\mathbf{1}\mathbf{1}^T$ is the centering matrix, $\mathbf{1} \in \mathbb{R}^n$ is a vector of all 1's, and $D^{(2)}_{\text{PM}}$ is obtained from D_{PM} by squaring all entries. We let the spectral decomposition of B be denoted by $B = V\Lambda V^T$, where $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_n)$, $V = (v_1, \ldots, v_n) \in \mathbb{R}^{n\times n}$ contain the eigenvalues and eigenvectors of B in descending order. The embedding dimension r is then chosen as the index i which maximizes the eigenratio λ_i/λ_{i+1} (40), with the following restrictions: we constrain $3 \leq i \leq 39$ and only consider ratios λ_{i+1}/λ_i between "large" eigenvalues, i.e. we require $\lambda_i/\lambda_1 \geq 0.01$. The PM-MDS embedding is then defined by $Y = (\sqrt{\lambda_1}v_1, \ldots, \sqrt{\lambda_r}v_r) \in \mathbb{R}^{n\times r}$.

Finally, we apply k-means to the PM-MDS embedding to obtain cluster labels. Specifically, we let $\hat{\ell}_i \in [k] = \{1, \ldots, k\}$ be the cluster label of x_i returned by running k-means on Y with k clusters and 20 replicates. Since k-means may return highly imbalanced clusters, cluster sample sizes were constrained to be at least $\sqrt{n}/2$. Specifically, if k-means returned a tiny cluster, k was increased to $k+1$, and the tiny cluster merged with the closest non-trivial cluster. This entire procedure is summarized in the pseudocode in Algorithm 1.

We note that the computational bottleneck for Algorithm 1 is the computation and storage of all pairwise path distances, which has complexity $O(n^2 \log n)$ when $K_2 = O(\log n)$. However this quadratic cost can be avoided by utilizing a low rank approximation of the squared distance matrix via the Nystrom method (41–45). For example, (46) propose a fast, quasi-linear implementation of MDS which only requires the computation of path distances from a set of q landmarks, so that the complexity of computing path distances is reduced to $O(qn \log n)$.

We also note that an important consideration in the fully unsupervised setting is how to select the number of clusters k. This is a rather ill-posed question with multiple reasonable answers due to hierarchical cluster structure. We do not focus on this in the current article, and Algorithm 1 assumes the number of clusters is given. However we emphasize that when k is unknown, the PM-MDS embedding offers a useful tool for selecting a reasonable number of clusters. For example, Line 21 of Algorithm 1 can be repeated for a range of candidate k values to obtain candidate clusterings $\widehat{\ell}_k$; $\widehat{k}$ can then be chosen so that $\widehat{\ell}_k$ optimizes a cluster validity criterion such as the silhouette criterion (47, 48). Alternatively, one could build a graph with distances computed in the PM-MDS embedding, and estimate k as the number of small eigenvalues of a corresponding graph Laplacian (25, 49).

Assessment. We evaluate the performance of Algorithm 1 with respect to (1) cluster quality and (2) geometric fidelity on a collection of labeled bench marking data sets with ground truth labels ℓ. There are many helpful metrics for the quality of the estimated cluster labels $\hat{\ell}$, and we compute the adjusted rand index (ARI), entropy of cluster accuracy (ECA), and entropy of cluster purity (ECP). Definitions of ECA and ECP can be found in Section 2 of the Supplemental. We compare our clustering results with the output of k-means, DBSCAN (50, 51), DBSCAN on UMAP embedding (14), and Seurat (10).

Assessing the geometric fidelity of the low-dimensional embedding Y is more delicate; we want to assess whether the embedding procedure preserves the global relative distances between clusters. We first compute the mean of each cluster as in (15) using the ground truth labels, i.e. $\mu_j(X) = \frac{1}{|\mathcal{I}_j|}\sum_{i\in\mathcal{I}_j} x_i$ where $\mathcal{I}_j = \{i : \ell_i = j\}$; we then define $D_{\mu,X}(i,j) = \|\mu_{\ell_i}(X) - \mu_{\ell_j}(X)\|_2$. Similarly, we compute the means $\mu_j(Y)$ in the PM-MDS embedding, and define $D_{\mu,Y}(i,j) = \|\mu_{\ell_i}(Y) - \mu_{\ell_j}(Y)\|_2$; we then compare $D_{\mu,X}$ and $D_{\mu,Y}$. Specifically, we define the geometric perturbation π by:

$$\pi(X, Y, \ell) = \min_c \frac{\left\|D_{\mu,X} - cD_{\mu,Y}\right\|_F^2}{\left\|D_{\mu,X}\right\|_F^2},$$

where $\|\cdot\|_F$ is the Frobenius norm. The c achieving the minimum is easy to compute, and one obtains

$$\pi(X,Y,\ell) = \frac{\left\|D_{\mu,X} - c^* D_{\mu,Y}\right\|_F^2}{\left\|D_{\mu,X}\right\|_F^2} \quad , \quad c* = \frac{\langle D_{\mu,X}, D_{\mu,Y}\rangle}{\left\|D_{\mu,Y}\right\|_F^2}.$$

We compare $\pi(X,Y,\ell)$ with the geometric perturbation of other embedding schemes for X, i.e. with $\pi(X,U,\ell)$ for U equal to the UMAP (14) and t-SNE (15) embeddings. Note that π is not always a useful measure: for example if X consisted of concentric spheres sharing the same center, the metric would be meaningless, as the distance between cluster means would be zero. Nevertheless, in most cases π is a helpful metric for quantifying the preservation of global cluster geometry.

Results

We apply Algorithm 1 to both a collection of toy manifold data sets and a collection of scRNAseq data sets. Results are reported in Sections and respectively. The default parameter values reported in Algorithm 1 were used on all data sets.

Manifold Data. We apply Algorithm 1 for $p = 1.5, 2, 4$ to the following four manifold data sets:

- **Balls** ($n = 1200$, $d = 2$, $k = 3$): Clusters were created by uniform sampling of 3 overlapping balls in $\mathbb{R}^2$; see Figure 1a.
- **Elongated with bridge** (denoted EWB, $n = 620$, $d = 2$, $k = 3$): Clusters were created by sampling from 3 elongated Gaussian distributions. A bridge was added connecting two of the Gaussians; see Figure 1b.
- **Swiss roll** ($n = 1275$, $d = 3$, $k = 3$): Clusters were created by uniform sampling from three distinct regions of a Swiss roll; 3-dimensional isotropic Gaussian noise ($\sigma = 0.75$) was then added to the data. Figure 1c shows the first two data coordinates.
- **SO(3) manifolds** ($n = 3000$, $d = 1000$, $k = 3$): For $1 \le i \le 3$, the 3-dimensional manifold $\mathcal{M}_i \subseteq \mathbb{R}^9$ is defined by fixing three eigenvalues $D_i = \text{diag}(\lambda_1, \lambda_2, \lambda_3)$ and then defining $\mathcal{M}_i = \cup_{V \in SO(3)} V D_i V^T$, where SO(3) is the special orthogonal group. After fixing D_i, we randomly sample from $\mathcal{M}_i$ by taking random orthonormal bases V of $\mathbb{R}^3$. A noisy, high-dimensional embedding was then obtained by adding uniform random noise with standard deviation $\sigma = 0.0075$ in 1000 dimensions. Figure 1d shows the first two principal components of the data, which exhibits no cluster separation.

The data sets were chosen to illustrate various cluster separability characteristics. For the balls, the clusters have good geometric separation but are not separable by density. For the Swiss roll and SO(3), the clusters have a complex and inter-twined geometry but are well separated in terms of density. For EWB, clusters are both elongated and lack robust density separability due to the bridge, and one expects that methods which rely too heavily on either geometry or density will fail. The ARIs achieved by Algorithm 1, k-means, DBSCAN, and Seurat are reported in Table 1. See Tables 2 and 1 in Section 2 of the Supplemental, for ECA and ECP. As expected, k-means out performs all methods on the balls but performs very poorly on all other data sets. DBSCAN and Seurat achieve perfect accuracy on the Swiss roll and SO(3) but perform rather poorly on the balls and EWB, although Seurat does noticeably better than DBSCAN. PM_2 is the only method which achieves a high ARI ($> 90\%$) on all data sets.

Table 2 reports the geometric perturbation of the embedding produced by Algorithm 1 and compares with UMAP and t-sne. Since Algorithm 1 generally selects an embedding dimension $r > 2$, to ensure a fair comparison the geometric perturbation was computed in both the 2d and r-dimensional (rd) embeddings for all methods, where r is the dimension selected by Algorithm 1. Overall $PM_{1.5}$ achieved the lowest geometric perturbation, although all methods had small perturbation on the Balls data set and UMAP had the lowest perturbation on EWB. We point out however that for both the Swiss roll and SO(3), the metric may not be meaningful due to the complicated cluster geometry.

scRNAseq Data. We apply Algorithm 1 for $p = 1.5, 2, 4$ to the following synthetic scRNAseq data sets:

- **Simulated beta:** Simulated data set of $n = 473$ beta cells and $d = 2279$ genes, created based on methods

Table 1. ARI for manifold data

Method	Balls	EWB	Swiss	SO(3)
k-means	**0.955**	-0.001	0.373	0.011
dbscan	0.056	0.552	1	1
UMAP+dbscan	0.648	0.515	1	1
Seurat	0.777	0.837	1	1
$PM_{1.5}$	**0.921**	0.489	1	0.501
PM_2	0.907	**0.990**	**1**	**1**
PM_4	0.781	0.584	**1**	**1**

Table 2. Geometric perturbation for manifold data. The rd embeddings were computed with an embedding dimension of $r = 5$ for the balls, EWB, Swiss roll and $r = 7$ for SO(3), which corresponded to the estimated dimension for both $PM_{1.5}$ and PM_2.

Method	Balls	EWB	Swiss	SO(3)
2d UMAP	0.001	**0.01**	0.107	0.178
rd UMAP	0.001	0.04	0.106	0.173
2d t-sne	**0**	0.054	0.641	0.321
rd t-sne	NA	NA	NA	0.165
2d $PM_{1.5}$	**0**	0.033	**0.002**	**0.103**
rd $PM_{1.5}$	**0**	0.023	0.011	0.154
2d PM_2	**0**	0.146	0.025	0.156
rd PM_2	**0**	0.068	0.025	0.179
2d PM_4	0.003	0.191	0.056	0.194
rd PM_4	0.004	0.157	0.056	0.194

of two imputation packages for scRNAseq datasets, namely SAVER (52) and scImpute (53). According to SAVER we use a real dataset (Baron's pancreatic) generated by (54) and filter it to keep high quality cells and genes. This filtered data set is called reference and represents the true expression of genes in the different cells. We then subset the reference data set to include only Beta cells. Following the procedure as in scImpute, we randomly choose 10% of the genes to operate as marker genes. Three cell types are then created by scaling up their mean expression. In particular, we split the cells to 3 clusters. For each cluster we scale up the mean expression of 1/3 of the marker genes. Lastly we want to simulate the drop out effect. The SAVER simulation suggests to multiply each cell by an efficiency loss constant drawn by Gamma(10,100). Using S to refer to the data matrix resulting from the above steps, the final simulated data X is obtained by letting X_{ij} be a random drawn from Poisson(S_{ij}).

- **RNA mixture:** Benchmarking scRNAseq data set from (55). RNAmix1 was processed with Sort-seq and has $n = 340$ cells and $d = 14224$ genes. RNAmix2 was processed with CEL-seq2 and has $n = 296$ cells and $d = 14687$ genes. For the creation of the two data sets, RNA was extracted in bulk for each of the following cell lines: H2228, H1975, HCC827. Then the RNA was mixed in 7 different proportions (each defining a ground truth cluster label), diluted to single cell equivalent amounts ranging from 3.75pg to 30pg, and processed using CEL-seq2 and SORT-seq. See here for Supplemental info including ground truth geometric structure.

In addition to the synthetic data, we evaluate the performance of Algorithm 1 on the following real scRNAseq data sets:

- **Cell mixture data set:** Another benchmarking data set from (55) consisting of a mixture of 5 cell lines created with 10x sequencing platform. The cell line identity of a cell is also its true cluster label. The data set consists of $n = 3822$ cells and $d = 11786$ genes; we removed multiplets, based on the provided metadata file.

- **Baron's pancreatic:** Human pancreatic data set generated by (54). After quality control and SAVER imputation, there are $d = 14738$ genes and $n = 1844$ cells. For analysis purposes cells that belong in a group with less than 70 members were filtered out. The cell types associated with each cell were obtained by an iterative hierarchical clustering method that restricts genes enriched in one cell type from being used to separate other cell types. The enriched markers in every cluster defined the cell type of the cells that belong in that cluster.

- **Tabula Muris data sets:** Mouse scRNAseq data for different tissues and organs (56). We select the pancreatic data (TM Panc) with $n = 1444$ cells and $d = 23433$ genes and the lung data (TM Lung) with $n = 453$ cells and $d = 23433$ genes. The characterization of cell types was done with an FACS-based full length transcript analysis.

- **PBMC4k data set:** This data set includes the gene expression of Peripheral Blood Mononuclear Cells. The raw data are available from 10X Genomics. After quality control, saver imputation, and removing the two smallest clusters, there are $d = 16655$ genes and $n = 4316$ cells in the dataset. The ground truth cell types are provided by SingleR annotation after marker gene verification in github.com/SingleR.

Details about the pre-processing of data sets can be found in Section 1 of the Supplemental. For the following UMAP and t-sne results, Linnorm normalization was applied without denoising, as this normalization gave the best results. Note the embedding dimension r selected by Algorithm 1 ranged from 3 to 7 for $PM_{1.5}$ and PM_2, and from 3 to 11 for PM_4. Table 3 reports the ARI achieved by Algorithm 1 and other methods; see Tables 4 and 3 in Section 2 of the Supplemental for ECA and ECP. Once again PM_2 exhibits the best overall performance, with a high ARI ($\geq 90\%$) on all data sets except TM lung and PBMC4K. The non path metric methods had a low ARI ($< 90\%$) on 4-6 of the 8 data sets, while $PM_{1.5}$ and PM_4 had a low ARI on 3 of the 8 data sets. These results indicate that incorporating both density-based and geometric information when determining similarity generally leads to more robust results for scRNA-seq data. Although the optimal balance depends on the data set (for example PBMC4K does best with $p = 4$, while TMLung does best with $p = 1.5$), path metrics with a moderate p exhibit the best performance across a wide range of data sets.

We also investigated whether we could learn the ground truth number of clusters by optimizing the silhouette criterion in the PM-MDS embedding, and compared this with the number of clusters obtained from Seurat using the default resolution; see Table 5 in Section 2 of the Supplemental. For 5 out of the 8 RNA data sets evaluated in this article (RNAMix1, RNAMix2, BaronPanc,PBMC4k and CellMix), this procedure yielded an estimate for k which matched the number of distinct annotated labels. On the other hand, Seurat correctly estimates the number of clusters only for 2 out of the 8 RNA data sets (RNAMix1 and TMLung).

Table 4 reports the geometric perturbation. We see that increasing p almost always increases the geometric perturbation, with $PM_{1.5}$ yielding the smallest geometric perturbation on all data sets except BaronPanc, where PM_2 has the smallest geometric perturbation. Although $PM_{1.5}$ is the clear

Table 3. ARI for RNA data

Method	RNA1	RNA2	TMLung	Beta	TMPanc	BaronPanc	PBMC4K	CellMix
k-means	0.939	0.786	0.848	0.926	0.885	0.901	0.747	1
dbscan	**0.952**	0.826	0.73	0.794	0.739	0.56	0.889	1
UMAP+db	0.939	0.793	0.654	0.944	0.892	0.973	0.82	1
Seurat	0.792	0.667	0.843	0.921	0.547	0.8	0.889	0.993
Seurat (def.)	0.714	0.785	0.764	0.907	0.796	**0.971**	0.975	1
$PM_{1.5}$	0.939	0.924	**0.890**	0.944	0.616	0.935	0.753	**1**
PM_2	0.939	**0.973**	0.845	**0.956**	**0.918**	0.948	0.757	**1**
PM_4	0.939	0.939	0.731	0.461	0.775	0.958	**0.978**	**1**

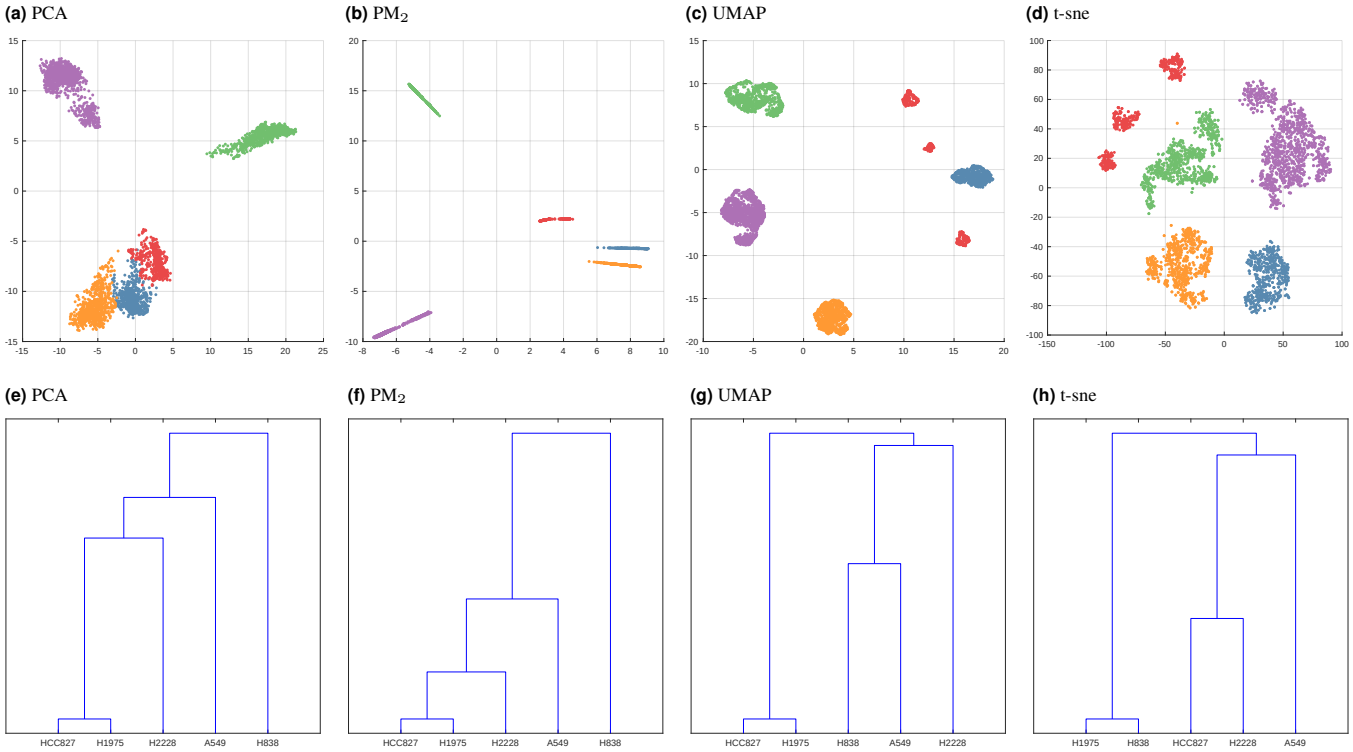

Fig. 4. Top row: PCA, PM_2, UMAP, and t-sne embeddings of Cell Mix data set, colored by true cell type. Bottom row: average linkage dendrograms of cluster means for the embeddings.

winner in terms of this metric, PM_2 still performed favorably with respect to UMAP and t-sne. Indeed, PM_2 had consistently lower geometric perturbation than UMAP, and when computed in r dimensions, it had lower geometric perturbation than t-sne except on the Tabula Muris data sets. Figure 4 shows the PCA, PM_2, UMAP, and t-sne embeddings of the Cell Mix data set, as well as a tree structure on the clusters. The tree structure was obtained by first computing the cluster means in the embedding and then applying hierarchical clustering with average linkage to the means. Interestingly path metrics recover the same hierarchical structure on the clusters as PCA: the cell types HCC827 and H1975 are the most similar, and H838 is the most distinct. This is what one would expect given more extensive biological information about the cell types, since H838 is the only cell line here derived from metastatic site Lymph node on a male patient, while both HCC827 and H1975 originated from the primary site of female lung cancer patients. However, neither UMAP or t-sne give the correct hierarchical representation of the clusters because both methods struggle to preserve global geometric structure as observed in numerous studies (57, 58).

Table 4. Geometric perturbation for RNA data. For rd UMAP and rd t-sne, $r = 7, 6, 5, 5, 5, 7, 3, 4$ for the various data sets, which was the maximum of the $PM_{1.5}$ dimension and the PM_2 dimension.

Method	RNA1	RNA2	TMLung	Beta	TMPanc	BaronPanc	PBMC4k	CellMix
2d UMAP	0.104	0.1	0.049	0.066	0.068	0.192	0.099	0.248
rd UMAP	0.140	0.068	0.18	0.057	0.041	0.129	0.070	0.035
2d t-sne	0.074	0.101	0.044	0.025	0.081	0.175	0.042	0.128
rd t-sne	0.055	0.043	0.027	0.046	0.034	0.154	0.040	0.017
2d $PM_{1.5}$	**0.010**	**0.013**	0.046	0.003	0.076	0.214	0.028	0.098
rd $PM_{1.5}$	0.017	0.009	**0.006**	**0.002**	**0.019**	0.040	**0.007**	**0.007**
2d PM_2	0.040	0.04	0.085	0.005	0.15	0.312	0.050	0.101
rd PM_2	0.048	0.036	0.029	0.005	0.051	**0.014**	0.013	0.008
2d PM_4	0.108	0.135	0.246	0.092	0.265	0.293	0.069	0.107
rd PM_4	0.100	0.082	0.083	0.013	0.099	0.044	0.029	0.008

Conclusions

This article applies a new theoretical framework to the analysis of single cell RNA-seq data which is based on the computation of optimal paths. Path metrics encode both geometric and density-based information, and the resulting low-dimensional embeddings simultaneously preserve density-based cluster structure as well as global cluster orientation. The method exhibits competitive performance when applied to numerous benchmarking data sets. Although we investigated other choices of p, we found that $p = 2$ performed well on a wide range of RNA data sets, indicating that $p = 2$ is an appropriate balance between density and geometry for this application. Future research will explore ways to make the method more robust to noise, tools for better visualization of the PM embeddings, and adapting the method to the semi-supervised context.

Funding

This work was supported in part by the NIH grants U01DE029255 and RO3DE027399, NSF grants DMS-1902906/DMS-2131292, and NSF Traineeship Program (DGE-1828149) to Andriana Manousidaki.

Software availability

Codes used for PM-MDS algorithm, comparison analysis and figure generation are available at scRNAseq_with_path-metrics github repository.

Data availability

The raw data of Cellmix and RNAmix are downloaded from GEO under accession code GSE118767, and the preprocessed data are available at their github repository. The simulated data are available at scRNAseq_with_path-metrics github page. The PBMC4K data is available at 10x Genomics's website. The Baron's pancreatic data is available in GEO with the access code GSM2230757. The mouse tissue scRNAseq data sets are accessible on Figshare.

Bibliography

1. Antoine-Emmanuel Saliba, Alexander J. Westermann, Stanislaw A. Gorski, and Jörg Vogel. Single-cell RNA-seq: advances and future challenges. *Nucleic Acids Research*, 42(14): 8845–8860, 07 2014. ISSN 0305-1048. doi: 10.1093/nar/gku555.
2. James Eberwine, Hermes Yeh, Kevin Miyashiro, Yanxiang Cao, Suresh Nair, Richard Finnell, Martha Zettel, and Paul Coleman. Analysis of gene expression in single live neurons. *Proceedings of the National Academy of Sciences*, 89(7):3010–3014, 1992.
3. Fuchou Tang, Catalin Barbacioru, Yangzhou Wang, Ellen Nordman, Clarence Lee, Nanlan Xu, Xiaohui Wang, John Bodeau, Brian B Tuch, Asim Siddiqui, et al. mrna-seq whole-transcriptome analysis of a single cell. *Nature methods*, 6(5):377–382, 2009.
4. Vladimir Yu Kiselev, Kristina Kirschner, Michael T Schaub, Tallulah Andrews, Andrew Yiu, Tamir Chandra, Kedar N Natarajan, Wolf Reik, Mauricio Barahona, Anthony R Green, and Martin Hemberg. SC3: consensus clustering of single-cell RNA-seq data. *Nature Methods*, 14:483–486, 05 2017. doi: 10.1038/nmeth.4236.
5. Josip S Herman, Dominic Grün, et al. Fateid infers cell fate bias in multipotent progenitors from single-cell rna-seq data. *Nature methods*, 15(5):379, 2018.
6. Peijie Lin, Michael Troup, and Joshua W. K. Ho. CIDR: Ultrafast and accurate clustering through imputation for single-cell RNA-seq data. *Genome Biology*, 18, 03 2017. doi: 10.1186/s13059-017-1188-0.
7. Zeisel A, Ana B Muñoz-Manchado, Simone Codeluppi, Lönnerberg P, La Manno G, Juréus A, Marques S, Munguba H, He L, Betsholtz C, Rolny C, Castelo-Branco G, Hjerling-Leffler J, and Linnarsson S and. Brain structure. Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. *Science (New York, N.Y.)*, 347:1138–1142, 03 2015. doi: 10.1126/science.aaa1934.
8. Justina Žurauskienė and Christopher Yau. pcaReduce: hierarchical clustering of single cell transcriptional profiles. *BMC Bioinformatics*, 17, 03 2016. doi: 10.1186/s12859-016-0984-y.
9. Jacob H. CLevine, Erin F. Simonds, Sean C. Bendall, Kara L. Davis, El-ad D. Amir, Michelle D. Tadmor, Oren Litvin, Harris G. Fienberg, Astraea Jager, Eli R. Zunder, Rachel Finck, Amanda L. Gedman, Ina Radtke, James R. Downing, Dana Pe'er, and Garry P. Nolan. Data-driven phenotypic dissection of aml reveals progenitor-like cells that correlate with prognosis. *Cell*, 2015. doi: doi:10.1016/j.cell.2015.05.047.
10. Chen Xu and Zhengchang Su. Identification of cell types from single-cell transcriptomes using a novel clustering method. *Bioinformatics*, 31(12):1974–1980, 2015.
11. Xiaoshu Zhu, Jie Zhang, Yunpei Xu, Jianxin Wang, Xiaoqing Peng, and Hong-Dong Li. Single-cell clustering based on shared nearest neighbor and graph partitioning. *Interdisciplinary Sciences: Computational Life Sciences*, 12:117–130, 2020. doi: 10.1007/s12539-019-00357-4.
12. Tim Stuart, Andrew Butler, Paul Hoffman, Christoph Hafemeister, Efthymia Papalexi, William M Mauck III, Yuhan Hao, Marlon Stoeckius, Peter Smibert, and Rahul Satija. Comprehensive integration of single-cell data. *Cell*, 177:1888–1902, 2019. doi: 10.1016/j.cell.2019.05.031.
13. F. Alexander Wolf, Philipp Angerer, and Fabian J. Theis. SCANPY: large-scale single-cell gene expression data analysis. *Genome Biology*, 19, 02 2018. doi: 10.1186/s13059-017-1382-0.
14. Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.
15. Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.
16. Kevin R. Moon, David van Dijk, Zheng Wang, Scott Gigante, Daniel B. Burkhardt, William S. Chen, Kristina Yim, Antonia van den Elzen, Matthew J. Hirn, Ronald R. Coifman, Natalia B. Ivanova, Guy Wolf, and Smita Krishnaswamy. Visualizing structure and transitions in high-dimensional biological data. *Nature Biotechnology*, 37(12):1482–1492, 2019. doi: 10.1038/s41587-019-0336-3.
17. Jian Tang, Jingzhou Liu, Ming Zhang, and Qiaozhu Mei. Visualizing large-scale and high-dimensional data. In *Proceedings of the 25th international conference on world wide web*, pages 287–297, 2016.
18. P. Vincent and Y. Bengio. Density-sensitive metrics and kernels. In *Snowbird Learning Workshop*, 2003.
19. O. Bousquet, O. Chapelle, and M. Hein. Measure based regularization. In *NIPS*, pages 1221–1228, 2004.
20. Sajama and A. Orlitsky. Estimating and computing density based distance metrics. In *ICML*, pages 760–767, 2005.
21. H. Chang and D.-Y. Yeung. Robust path-based spectral clustering. *Pattern Recognition*, 41 (1):191–203, 2008.
22. A.S. Bijral, N. Ratliff, and N. Srebro. Semi-supervised learning with density based distances. In *UAI*, pages 43–50, 2011.
23. A. Moscovich, A. Jaffe, and B.Nadler. Minimax-optimal semi-supervised regression on unknown manifolds. In *AISTATS*, pages 933–942, 2017.
24. D. Mckenzie and S. Damelin. Power weighted shortest paths for clustering Euclidean data. *Foundations of Data Science*, 1(3):307, 2019.
25. A. Little, M. Maggioni, and J.M Murphy. Path-based spectral clustering: Guarantees, robustness to outliers, and fast algorithms. *Journal of Machine Learning Research*, 21(6):1–66, 2020.
26. E. Borghini, X. Fernández, P. Groisman, and G. Mindlin. Intrinsic persistent homology via density-based metric learning. *arXiv preprint arXiv:2012.07621*, 2020.
27. Bernd Fischer, Thomas Zöller, and Joachim M Buhmann. Path based pairwise data clustering with application to texture segmentation. In *International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 235–250. Springer, 2001.
28. S. Zhang and J.M. Murphy. Hyperspectral image clustering with spatially-regularized ultrametrics. *Remote Sensing*, 13(5):955, 2021.
29. M. Pollack. Letter to the editor: The maximum capacity through a network. *Operations Research*, 8(5):733–736, 1960.
30. T.C. Hu. Letter to the editor: The maximum capacity route problem. *Operations Research*, 9(6):898–900, 1961.
31. P.M. Camerini. The min-max spanning tree problem and some extensions. *Information Processing Letters*, 1(10-14), 1978.
32. H. Gabow and R.E. Tarjan. Algorithms for two bottleneck optimization problems. *Journal of Algorithms*, 9:411–417, 1988.
33. John M Lee. *Introduction to Riemannian manifolds*. Springer, 2018.
34. Timothy Chu, Gary Miller, and Donald Sheehy. Exploration of a graph-based density sensitive metric. *arXiv preprint arXiv:1709.07797*, 2017.
35. Anna Little, Daniel McKenzie, and James Murphy. Balancing geometry and density: Path distances on high-dimensional data. *arXiv preprint arXiv:2012.09385*, 2020.
36. P. Groisman, M. Jonckheere, and F. Sapienza. Nonhomogeneous Euclidean first-passage percolation and distance learning. *arXiv preprint arXiv:1810.09398*, 2018.
37. T. Chu, G.L. Miller, and D.R. Sheehy. Exact computation of a manifold metric, via Lipschitz embeddings and shortest paths on a graph. In *SODA*, pages 411–425, 2020.
38. N. García Trillos, D. Sanz-Alonso, and R. Yang. Local regularization of noisy point clouds: Improved global geometric estimates and data analysis. *Journal of Machine Learning Research*, 20(136):1–37, 2019.
39. Ingwer Borg and Patrick JF Groenen. *Modern multidimensional scaling: Theory and applications*. Springer Science & Business Media, 2005.
40. Clifford Lam and Qiwei Yao. Factor modeling for high-dimensional time series: inference for the number of factors. *The Annals of Statistics*, pages 694–726, 2012.
41. Christopher Williams and Matthias Seeger. Using the nyström method to speed up kernel machines. In *Proceedings of the 14th annual conference on neural information processing systems*, number CONF, pages 682–688, 2001.
42. Benyamin Ghojogh, Ali Ghodsi, Fakhri Karray, and Mark Crowley. Multidimensional scaling, sammon mapping, and isomap: Tutorial and survey. *arXiv preprint arXiv:2009.08136*, 2020.
43. John Platt. Fastmap, metricmap, and landmark mds are all nyström algorithms. In *International Workshop on Artificial Intelligence and Statistics*, pages 261–268. PMLR, 2005.
44. H Yu, X Zhao, X Zhang, and Y Yang. Isomap using nyström method with incremental sampling. *Advances in Information Sciences & Service Sciences*, 4(12), 2012.
45. Ali Civril, Malik Magdon-Ismail, and Eli Bocek-Rivele. Ssde: Fast graph drawing using sampled spectral distance embedding. In *International Symposium on Graph Drawing*, pages 30–41. Springer, 2006.
46. Gil Shamai, Michael Zibulevsky, and Ron Kimmel. Efficient inter-geodesic distance computation and fast classical scaling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(1):74–85, 2020. doi: 10.1109/TPAMI.2018.2877961.
47. Leonard Kaufman and Peter Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis*. 09 2009. ISBN 9780470317488.
48. Martin Maechler, Peter Rousseeuw, Anja Struyf, Mia Hubert, and Kurt Hornik. *cluster: Cluster Analysis Basics and Extensions*, 2021.
49. Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.
50. Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231, 1996.
51. Xiaowei Xu, Martin Ester, H-P Kriegel, and Jörg Sander. A distribution-based clustering algorithm for mining in large spatial databases. In *Proceedings 14th International Conference on Data Engineering*, pages 324–331. IEEE, 1998.
52. Mo Huang, Jingshu Wang, Eduardo Torre, Hannah Dueck, Sydney Shaffer, Roberto Bonasio, John I Murray, Arjun Raj, Mingyao Li, and Nancy R Zhang. Saver: gene expression recovery for single-cell rna sequencing. *Nature methods*, 15(7):539–542, 2018.
53. Wei Vivian Li and Jingyi Jessica Li. An accurate and robust imputation method scimpute for single-cell rna-seq data. *Nature communications*, 9(1):1–9, 2018.
54. Maayan Baron et al. A single-cell transcriptomic map of the human and mouse pancreas reveals inter- and intra-cell population structure. *Cell Systems*, 3(4):346–360, 2016.
55. Luyi Tian, Xueyi Dong, Saskia Freytag, Kim-Anh Lê Cao, Shian Su, Abolfazl JalalAbadi, Daniela Amann-Zalcenstein, Tom S Weber, Azadeh Seidi, Jafar S Jabbari, et al. Benchmarking single cell rna-sequencing analysis pipelines using mixture control experiments. *Nature methods*, 16(6):479–487, 2019.
56. Logistical coordination et al. Tabula Muris Consortium, Overall coordination. Single-cell

transcriptomics of 20 mouse organs creates a tabula muris. *Nature*, 562:367–372, 2018.
57. Dmitry Kobak and Philipp Berens. The art of using t-sne for single-cell transcriptomics. *Nature Communications*, 10:2041–1723, 2019. doi: 10.1038/s41467-019-13056-x.
58. Shamus M. Cooley, Timothy Hamilton, J. Christian J. Ray, and Eric J. Deeds. A novel metric reveals previously unrecognized distortion in dimensionality reduction of scrna-seq data. *bioRxiv*, 2020. doi: 10.1101/689851.