A novel metric reveals previously unrecognized distortion in dimensionality reduction of scRNA-seq data.

Shamus M. Cooley[1,3,4,+] , Timothy Hamilton[1,3,+], Samuel D. Aragones[3], J. Christian J. Ray[4,5,6*], and Eric J. Deeds[2,3,4,*]

1. Interdepartmental Program for Bioinformatics, University of California, Los Angeles, CA

2. Department of Integrative Biology and Physiology, University of California, Los Angeles, CA

3. Institute for Quantitative and Computational Biosciences, University of California, Los Angeles, CA

4. Center for Computational Biology, University of Kansas, Lawrence, KS

5. Department of Molecular Biosciences, University of Kansas, Lawrence, KS

6. Current Address: Inscripta, Boulder, CO

*Corresponding authors: JCJR (christian.ray@inscripta.com) and EJD (deeds@ucla.edu)

+These authors contributed equally to this work

Abstract

High-dimensional data are becoming increasingly common in nearly all areas of science. Developing approaches to analyze these data and understand their meaning is a pressing issue. This is particularly true for single-cell RNA-seq (scRNA-seq), a technique that simultaneously measures the expression of tens of thousands of genes in thousands to millions of single cells. The emerging consensus for analysis workflows significantly reduces the dimensionality of the dataset before performing downstream analysis, such as assignment of cell types. One problem with this approach is that dimensionality reduction can introduce substantial distortion into the data; consider the familiar example of trying to represent the three-dimensional earth as a two-dimensional map. It is currently unclear if such distortion affects analysis of scRNA-seq data. Here, we introduce a straightforward approach to quantifying this distortion by comparing the local neighborhoods of points before and after dimensionality reduction. We found that popular techniques like t-SNE and UMAP introduce substantial distortion even for relatively simple simulated data sets. For scRNA-seq data, we found the distortion in local neighborhoods was often greater than 95% in the representations typically used for downstream analyses. This level of distortion can introduce errors into cell type identification, pseudotime ordering, and other analyses. We found that principal component analysis can generate accurate embeddings, but only when using dimensionalities that are much higher than typically used in scRNA-seq analysis. Our work suggests the need for a new generation of dimensional reduction algorithms that can accurately embed high dimensional data in its true latent dimension.

Introduction

Technological advances over the past century have enabled collection and analysis of data sets of unprecedented size and complexity. In geology, a modern assay might report the concentrations for over fifty elements from a single sample (Horrocks et al. 2019). In climatology, measurements of sea surface temperature and the strength of zonal winds can be obtained simultaneously from hundreds of different sensors at any given point in time (Chalupka et al. 2016). In cell and molecular biology, sequencing technologies have scaled up the throughput and resolution of genome data in populations (Lemmon and Lemmon 2013; Ozsolak and Milos 2011) and gene expression levels in single cells (Lake et al. 2018; Stegle et al. 2015), into many thousands of dimensions in the case of single cell RNA-seq (scRNA-seq). Future technologies will doubtlessly expand the numbers of dimensions detected in complex systems by orders of magnitude.

While such datasets promise to provide greater insight into the problems being studied, high-dimensional data are also more difficult to analyze. The computational complexity of many data analysis algorithms scales exponentially with the dimensionality of the dataset, statistical inference often becomes difficult as dimensionality increases, and algorithms that work in lower dimensions become intractable in higher-dimensional spaces (Indyk and Motwani 1998; Friedman 1997). This is often referred to as the “curse of dimensionality”. The aim of dimensionality reduction is to reduce the scale of the problem while retaining as much of the relevant information as possible–ideally all of it. It has become an indispensable tool for the rapidly growing number of scRNA-seq studies.

Dimensionality reduction has a long history (Pearson 1901; Hotelling 1933). Principal Component Analysis (PCA) is perhaps the oldest and most common linear approach, but many alternative approaches to linear dimensionality reduction exist as well, such as Non-negative Matrix Factorization (NMF) and Independent Component Analysis (ICA) (Pearson 1901; Cichocki and Phan 2009). These algorithms are useful in a broad class of problems. However, linear approaches may be insufficient when the data display significant nonlinear characteristics (DeMers and Cottrell 1993). In such situations, one often adopts a "manifold" assumption, which posits that the data can be modeled as smoothly varying local neighborhoods of dimension significantly lower than the ambient space (Moon et al. 2018). A large number of Nonlinear Dimensionality Reduction (NDR) techniques have been developed to approximate these manifolds (Tenenbaum et al. 2000; Kruskal 1964; Knyazev 1998; Roweis and Saul 2000), including popular visualization methods like t-distributed Stochastic Neighbor Embedding (t-SNE) (Laurens van der Maaten et al. 2008) and Uniform Manifold Approximation and Projection (UMAP) (McInnes et al. 2018). Collectively, the use of NDR techniques is often referred to as "manifold learning" (Moon et al. 2018).

In all dimensionality reduction techniques, one specifies the dimension of the resulting representation of the data. For example, if we use t-SNE to reduce the dimension of scRNA-seq data, we tell the algorithm the number of dimensions that we want in the end. Unfortunately, the appropriate (or *latent*) dimensionality needed to correctly represent any given data set is generally not known *a priori*. A natural choice for visualization purposes is to choose two dimensions, since that kind of representation is easy to reproduce in the format of a figure. In the analysis of scRNA-seq data, two dimensions are sometimes used not just for visualization but also for downstream analyses ranging from cell type clustering (Fig. 1a) to "pseudotime" ordering (Trapnell et al. 2014a). Currently, it is unclear just how much character of the original data is being lost in the reduction of data on the order of 20,000 dimensions, typical for scRNA-seq in many species, to two dimensions. Even when more dimensions are employed to represent the data, the amount of information preserved in the dimensionality reduction step is not obvious.

In order to understand the issues that might be introduced through dimensionality reduction, consider the familiar problem of making a 2-D map of the entire surface of the Earth. Doing this requires "slicing" the earth along some axis to unfold it into a map; this is commonly done in a line through the Pacific, since few landmasses are disrupted by this cut. Then, the mapmaker must either increase the relative size of landmasses near the poles or slice the map again to project the globe into two dimensions. Regardless of technique, the globe cannot be represented in two dimensions without slicing and distorting the map in some way, which has led, for instance, to popular criticisms of the Mercator Projection. While distortion of distance and area are of course important, perhaps more concerning is the fact that the discontinuous slices mentioned above take points that are nearby (e.g., two points in the Pacific) and place them on opposite sides of the map. This means that the local neighborhoods of many of the points on the globe are completely different between the Earth itself and the 2-D representation.

With this observation in mind, it becomes apparent that there is no guarantee that high dimensional data sets, such as those associated with single cell genomics, can be represented in two dimensions without introducing analogous discontinuous slices into the data. Even techniques that attempt to objectively find a lower-dimensional representation using more than two dimensions, such as the common "scree" (or "elbow") plot technique in PCA to choose the directions that capture most of the variation in the data (Cattell 1966), could also suffer from similar problems. Yet, little analysis has been done to elucidate the extent to which NDR techniques introduce discontinuities into reduced-dimensional representations.

We approached this problem by applying a simple metric, inspired by the above metaphor of the globe, to quantify the extent to which any given dimensionality reduction technique discontinuously slices or folds the data in some way. This metric is based on comparing the *local neighborhood* of a point in the original data with the local neighborhood of that same point in the reduced-dimensional space using the Jaccard distance (Levandowsky and Winter 1971). We first applied this approach to the simple problem of embedding points on the surface of a hypersphere (which is a straightforward generalization of the sphere to more than three dimensions) into the appropriate latent dimension from a higher-dimensional space. We found that many popular techniques, such as t-SNE and UMAP, not only introduced discontinuous slices into the data when trying to embed hyperspheres into two dimensions, but also when trying to embed into the correct latent dimension. Indeed, we failed to identify an NDR technique currently in widespread use for analysis or visualization of scRNA-seq data that could successfully embed hyperspheres above approximately 10 dimensions. We found similar results with other types of simulated data sets that are typically used to represent scRNA-seq data; none of the popular NDR techniques could successfully embed any of these data sets, even in the known latent dimension.

We then used our metric to analyze how dimensionality reduction affects analysis of scRNA-seq data. While nearly every published analysis of scRNA-seq data uses a unique set of steps and parameters, common approaches entail several steps of dimensionality reduction. In the first step, a set of “Highly Variable Genes” (HVGs) are selected from the data set, usually by identifying those genes whose variance in the data set is higher than would be expected based on the average expression level of that gene (Luecken and Theis 2019; Andrews et al. 2021). While there are some rough guidelines available in the literature, the choice of the number of HVGs to use for downstream analysis is ultimately arbitrary (Luecken and Theis 2019; Andrews et al. 2021). The HVGs are then used as input to PCA in order to represent the data in a lower-dimensional space. The number of principal components chosen for this step is also arbitrary but is often based on visual inspection of a scree plot. Finally, the data are visualized in 2 or 3 dimensions using NDR tools, most often t-SNE or UMAP (Laurens van der Maaten et al. 2008; McInnes et al. 2018   ). In reviewing the literature, one finds that some groups perform further quantitative analysis on the higher-dimensional PCA representation of the data (Siebert et al. 2019; Cao et al. 2019; Davie et al. 2018), while others perform analysis on the 2 or 3 dimensional embeddings given by UMAP or t-SNE (Rosenberg et al. 2018; Jean-Baptiste et al. 2019; Taylor et al. 2019).

Using our AJD metric, we found that each of these steps introduces tremendous distortion into the data. Specifically, commonly used pipelines disrupt 90-99% of the local neighborhoods in the data *prior* to performing further quantitative analysis. Like our findings on simulated data, even when embedding into higher dimensions, NDR techniques generally introduced substantial discontinuity into the data. While PCA can find embeddings with relatively low levels of distortion, this can generally only be achieved at much higher dimensionalities than are typically used. Interestingly, it has been suggested that the PCA step “de-noises” the data by finding “true” directions of variance and ignoring directions of variance that correspond to “noise” (Wagner et al. 2019). We performed an extensive analysis of the effect of PCA on simulated datasets and found that PCA could not successfully remove noise in local neighborhoods unless the levels of noise were extremely small.

Overall, our findings demonstrate that, regardless of the linear or non-linear technique used to reduce dimensionality, most of the local structure of high-dimensional data is lost when compressed into the number of dimensions typically used for scRNA-seq analysis. Indeed, our results indicate that any analysis based on lower-dimensional representations of the data can introduce substantial bias into interpretations of the results. Furthermore, we show that the distortion introduced by NDR techniques applied to existing scRNA-seq datasets can significantly alter the results of downstream analyses like cell type clustering and pseudotime ordering. Our findings suggest straightforward guidelines for evaluating the quality of a lower-dimensional representation of scRNA-seq data. Nevertheless, new NDR techniques are needed that can reliably produce true topological embeddings, or, at least, closer approximations than current techniques can produce. We expect that the metric and approach introduced here will be helpful in evaluating and developing more effective approaches to the problem of manifold learning and analysis of scRNA-seq or other high-dimensional data.

Results

Quantifying discontinuities introduced by dimensionality reduction

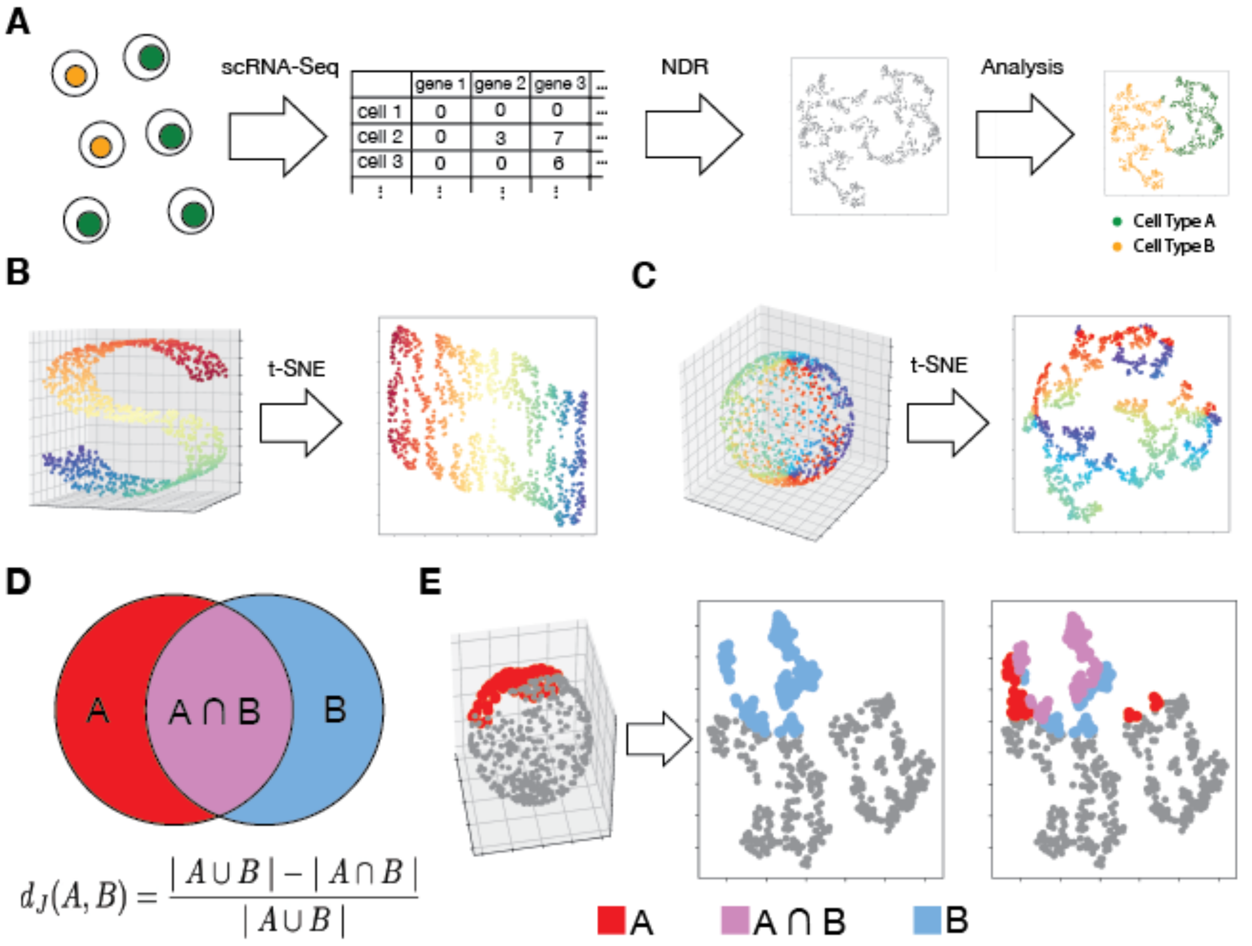
The goal of NDR is to learn a representation of a data set that has fewer features, but still retains the bulk of the information contained in the data. The extent to which the representations created by dimensionality reduction techniques preserve information is often illustrated with toy datasets such as the swiss roll (Fig. 1b). This example tests the ability of NDR techniques to represent the three-dimensional swiss roll data set in two dimensions while preserving the local structure of the original dataset (as can be seen here by the preservation of the "rainbow" pattern in the t-SNE representation). Most NDR techniques perform well on this task because a swiss roll is just a "rolled up" two-dimensional plane – a relatively simple transformation of a plane into a three-dimensional object. However, many objects, like the sphere in Fig. 1c, cannot be represented in 2-D without introducing significant distortion in local neighborhoods. This results in a notable scattering of the rainbow pattern (Fig. 1c).

Mathematically, a mapping from a high dimension to a lower dimension that (locally) preserves the structure of the data is called an *embedding*: technically, this a bijective map that is continuous in both directions (also called a *homeomorphism*). For topological spaces, a key mathematical property of an embedding is that it is *continuous*, and a consequence of that continuity is that local neighborhoods (e.g., the rainbow pattern in Fig. 1c) are preserved. For a swiss roll, NDR techniques like t-SNE can usually find an embedding, or something close to one. For a sphere, however, NDR finds a representation of the data in two dimensions that is not, strictly speaking, an embedding.

It is clear from the simple example in Fig. 1c that a major problem with trying to embed a sphere in 2-D is that this is impossible to do without introducing discontinuities into the resulting representation. In the context of experimental scRNA-seq data, this means that the local structure of the data may be lost in the dimensionality reduction, and errors (possibly large errors) could be introduced into any analysis that happens downstream of dimensionality reduction. This is particularly problematic because we do not know *a priori* what the true dimension of a particular scRNA-seq data set might be. Previous work on quantifying distortion in NDR has focused on the notion of Euclidean distance between the position of a point in the original space and its embedded position or the distances between points (McInnes et al. 2018 ; Zhang and Zha 2004), without considering the change in relative position between the point and its neighbors. However, quantifying the extent of the loss of structure caused by NDR requires consideration of neighborhoods within the data, not just changes in the positions of individual points. For example, a 2-D representation of the swiss roll might be stretched out, greatly distorting the pointwise distances, while still maintaining the rainbow structure depicted in Fig. 1c and thus providing a true embedding. This suggests the need to develop alternative approaches to quantifying distortion in NDR, particularly focused on characterizing discontinuities that may be introduced by dimensionality reduction techniques.

For any point in the swiss roll, the neighborhood of other points that are nearest to it are roughly the same in three dimensions and in the t-SNE representation in two dimensions (Fig. 1b). The two-dimensional representation of the sphere, on the other hand, gives noticeably different sets of nearest neighbors to many points (Fig. 1c). We thus developed a straightforward metric based on quantifying how similar the sets of neighbors are around each point between the original, high-dimensional data in the ambient space, and the low-dimensional representation. First, we find the *k*-nearest neighbors for each point in the original data. We call this set A (see Fig. 1d). Next, we find the *k*-nearest neighbors of that same point in the lower-dimensional space. We call this set B. We compare these two sets using a measure of dissimilarity called the Jaccard distance (Fig. 1e). Calculating the Jaccard distance involves computing the size (or *cardinality*) of the *symmetric difference* between A and B: the symmetric difference is just the set of points that are in A or B, but not both. This is equivalent to subtracting the number of points in the intersection between A and B from the number of points in the union (Fig. 1e). The Jaccard distance is the ratio of the size of this symmetric difference to the total number of points in A and B together (i.e. the number of points in the union between A and B).

If A and B are identical sets, meaning the neighbors of the point in the high-dimensional data and the low-dimensional representation are the same, then the Jaccard distance is 0. If A and B are completely different sets (i.e., the neighbors around this point completely change) then the Jaccard distance is 1. It is easy to show that, for a true topological embedding the Jaccard distance will be zero for every point in the dataset (Supplemental Info); in other words, in a true embedding all local information is preserved. To characterize the global "distance" of any low-dimensional representation from this ideal, we first compute the Jaccard distance for all the points in the data set and then average these values. We refer to this quantity as the Average Jaccard Distance (AJD), and it gives a value of 0 for a true embedding, 1 for a representation that retains none of the information about the local structure of the data for any point in the data set, and an intermediate value for a representation that retains part of the information.



	gene 1	gene 2	gene 3	...
cell 1	0	0	0	...
cell 2	0	3	7	...
cell 3	0	0	6	...
⋮	⋮	⋮	⋮	

$$d_J(A,B) = \frac{|A \cup B| - |A \cap B|}{|A \cup B|}$$

Fig. 1. Measuring distortion in dimensionality reduction (A) A schematic of some scRNA-seq workflows. The gene expression data are stored as a matrix, with each row corresponding to a cell, and each column correspond to a gene (after correcting for UMI swapping). The data undergo dimensionality reduction, and analysis is performed on the lower-dimensional representation of the data. **(B)** The "swiss roll" data set. t-SNE can reduce the data into two dimensions without altering the local structure of the data. **(C)** A sphere data set. t-SNE is

unable to represent the 3-dimensional object in 2 dimensions without disrupting the local structure of the data. **(D)** The Jaccard Distance is a method for quantifying the disruption in local neighborhoods pictured in . **(E)** An illustration of how NDR distorts local neighborhoods. The red points are the *k*-nearest neighbors of a single point in the 3-dimensional space. The blue points are the *k*-nearest neighbors of the same point in the t-SNE-generated 2-dimensional representation. The violet points are the intersection between the red points and the blue points. Thus the number of neighbors that that are not preserved as result of the embedding can be

Testing on Synthetic Data

To test the usefulness of the AJD, we first applied the metric to a problem where we know *a priori* the appropriate embedding dimension for the data set. Specifically, we created synthetic data for hyperspheres of varying dimension. A hypersphere is a manifold that represents a straightforward generalization of the standard 3-dimensional sphere to higher numbers of dimensions; it is just a collection of points in some *n*-dimensional space that are all the same distance from a central point (that distance is the radius of the sphere). In two dimensions this is a circle, in three dimensions a sphere, and in higher dimensions a hypersphere. We used a simple algorithm to sample uniformly from the surface of a hypersphere in *n* dimensions; for simplicity we used the origin of the space as the central point, and we set the radius of the hypersphere to 1 (see Methods). It is mathematically impossible to embed an *n*-dimensional sphere generated this way in less than *n* dimensions, so we called *n* the "latent dimension" of the data. To see if NDR techniques could generate a true embedding of the data into *n* dimensions, we first embedded our hyperspheres into a 100-dimensional ambient space. To demonstrate how we did this, take the case of a 20-dimensional hypersphere. If we sample points from that hypersphere, each one of those points is characterized by a vector of 20 numbers. We can trivially embed those points into a 100-dimensional space by just adding 80 zeroes to the end of those vectors (see Methods and Supporting Info).

We used the approach above to generate synthetic 100-dimensional datasets with 1000 points sampled from hyperspheres of known latent dimension. We then used multiple NDR techniques to embed this dataset into each lower dimension from 1 to 100. We hypothesized that the AJD would be zero for every dimension above the latent dimensionality n of the manifold that we had generated. Surprisingly, however, we found that the AJD did not reach 0 for hyperspheres with $n \geq 10$ for any NDR technique that we tried when we used a neighborhood size of $k = 20$ (see Fig. 2a). In the case of the popular technique t-SNE, for instance, the embeddings it produced generally had AJDs of greater than 0.75, regardless of both the latent dimension of the hypersphere and the embedding dimension used for the t-SNE algorithm. Other techniques, such as Isomap and Spectral Embedding (DeMers and Cottrell 1993; Tenenbaum et al. 2000) exhibited clear minima in the AJD at the appropriate latent dimension, but still produced embeddings with significant distortion. Changing the size of the neighborhood between 10 and 100 points did not significantly alter these findings (Supplemental Figure 2). This result is particularly striking because we know that it is possible to embed a 20-dimensional hypersphere into a 20-dimensional space without any distortion at all (corresponding to an AJD of 0). Indeed, for the case of this synthetic dataset there is a trivial mapping that results in a true embedding and an AJD of zero in the latent dimension, but none of the commonly used techniques that we tested successfully recovered it.

Of course, while a hypersphere is a very classic form of smooth manifold, such structures certainly do not represent a good approximation for the structure of scRNA-seq data. We thus considered several other types of simulated data where we could define the latent dimensionality unambiguously. The first example was sampling from a simple multivariate Gaussian, which is sometimes used as a model for scRNA-seq data (Zappia et al. 2017; Papadopoulos et al. 2019).

Not surprisingly, NDR techniques showed similar performance for a 20-dimensional Gaussian and a 20-dimensional hypersphere (Fig. 2C). We also used the popular “Splatter” package to simulate scRNA-seq data (Zappia et al. 2017); this approach simulates the existence of multiple “cell types” and more accurately attempts to replicate the structure of scRNA-seq data. We used Splatter to simulate a scenario with 5 cell types and 20 different genes, and again trivially embedded the simulated data in a 100-dimensional space. As in the case of the more simplified manifolds discussed above, none of the NDR techniques we tried, including the popular t-SNE and UMAP tools, could successfully embed the data, even in the known latent dimension (Fig. 2C).

We hypothesized that the results described above may have been because the datasets were too small, and that an increased sample size might allow the algorithms to find a proper embedding. Although increasing the sample size created a more pronounced local minimum at the latent dimension for some techniques (Fig. 2b), the AJD at the latent dimension never dropped below a certain level: this minimum was (essentially) invariant to increases in sample size of points on the sphere up to 5000 points (Figs. 2D). In the case of MDS, increasing sample size resulted in *more* distorted representations at the latent dimension. Many recent scRNA-seq studies have been able to capture data for 10s of thousands to millions of cells, so it might be that NDR techniques can successfully embed data sets once the number of points approaches that size. Given the computational costs of NDR with large sample sizes, we only tested PCA and UMAP for these large sample sizes. Interestingly, while PCA can recover the 20-dimensional embedding for these hypersphere data sets without any distortion regardless of sample size, massive increases in the number of points only modestly improved UMAP performance. Although in theory UMAP might be able to obtain a distortion-free embedding with truly

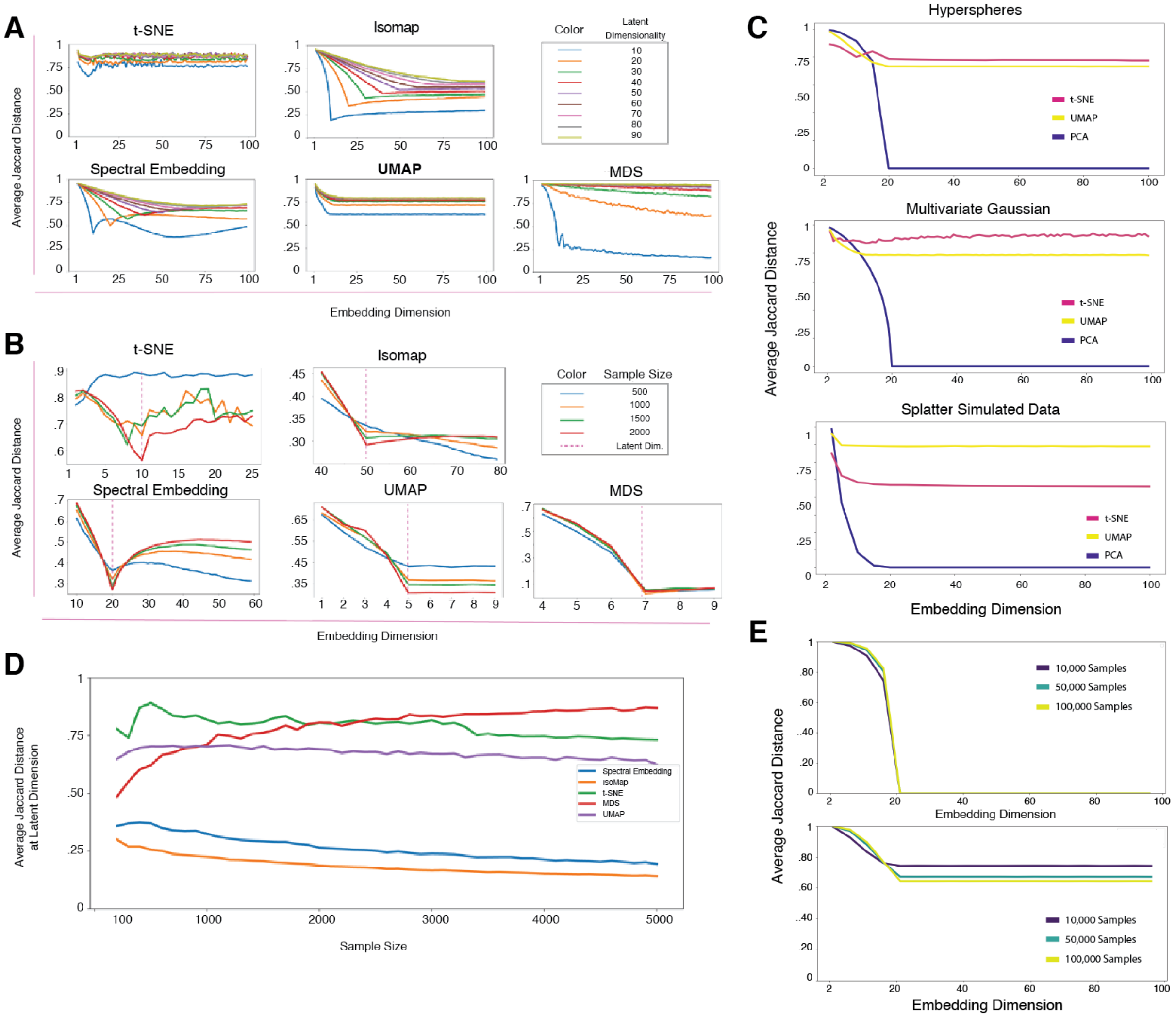


Fig. 2. Distortion in simulated data (A) The Average Jaccard Distance (AJD) for points randomly sampled from the surface of hyperspheres of varying dimension embedded in dimensions 1-100. We found that the best AJD is lowest when the latent dimensionality of the manifold is lowest, but these NDR techniques uniformly fail to find low-distortion embeddings. **(B)** The effect of sample size on Average Jaccard Distance. Although the shape of the curve more clearly indicates the latent dimensionality of the manifold, the distortion in local structure (AJD) does not improve with increased sample size. **(C)** AJD for varying high-dimensional geometries. Three simulated 20-Dimensional datasets, hyperspheres, multivariate gaussians, and virtual scRNAseq data simulated by the Splatter package, are each embedded into spaces of dimension varying from 2-100. The AJD is calculated for each embedding. **(D)** AJD vs. Sample size. The Average Jaccard Distance as the sample size increases from 100-5000 points. The distortion created by the embedding is mostly independent of sample size. (The latent dimension of these datasets was 20, and the ambient dimension of these datasets was 100.) **(E)** Large Sample Sizes. Datasets are sampled from a 20-dimensional hypersphere and embedded in

spaces of varying dimension. Increase the size of the sample does not alleviate the distortion introduced by dimensionality reduction.

massive data sets (say, 10s of millions of points), the method is too slow to be used at such sample sizes, indicating that UMAP cannot generate low-distortion embeddings in practice. Again, these simulated datasets represent what should be a relatively trivial problem for manifold learning. The fact that no nonlinear dimensionality reduction technique could find even this simple mapping raises questions about the accuracy of the approximate "embeddings" generated by NDR and the effects that distortion might have on the analysis of scRNA-seq and other high-dimensional data.

Measuring Distortion in scRNA-seq Studies

Our work on synthetic data suggests that NDR techniques cannot learn distortion-free embeddings of data even when the manifolds in question are relatively simple. It is thus unclear how much distortion common pipelines introduce into lower-dimensional representations of the data. To address this question, we identified a large set of state-of-the-art scRNA-seq studies (Siebert et al. 2019; Cao et al. 2019) and analyzed the effect of NDR on the analysis of these data. First, we looked at a study of Hydra cells by Siebert et al.2019; we focused on this data set because it was relatively large (~24,000 cells) and captures the range of transcriptional variation present across all cell types in a complex animal. We then considered how much distortion was introduced into the 2-D t-SNE and UMAP representations of the data, following a typical dimensionality reduction workflow. Specifically, we first selected 5000 HVGs for all the cells in the data set using standard tools in the popular python-based scRNA-seq analysis package Scanpy (Wolf et al. 2018), and then reduced the dimensionality of this subset with PCA using 45 principal components (the number of PCs was selected by visual inspection of a scree plot). We then reduced the data down to 2 dimensions using both t-SNE and UMAP. We found that both tools introduced massive levels of distortion into the data, with AJD values of around 0.9 (Fig. 3A). To determine how NDR tools would perform on smaller, more focused datasets, we selected one of the largest cell type clusters defined in the study (1,778 cells), an endodermal epithelial stem cell, and performed the same analysis on this subset of cells as we had for the entire dataset. While UMAP was able to find a lower-distortion embedding for this smaller dataset, t-SNE had worse performance, and both techniques had AJD values over 0.75 (Figs. 3A and B). This result suggests that, while 2-D visualizations of scRNA-seq datasets can be helpful for summarizing the data, they generally generate local neighborhoods that are almost *completely distinct* from those present in the original data. It is important to note that many recent studies perform their downstream analyses like cell type clustering on the t-SNE or UMAP representation of the data (Rosenberg et al. 2018; Jean-Baptiste et al. 2019; Taylor et al. 2019; Zhong et al. 2018); such an approach likely involves analysis of data with extremely distorted local neighborhoods, which may strongly impact the results (as discussed below).

Many analyses focus on the PCA representation of the data, however, rather than the 2-D t-SNE or UMAP representation (Siebert et al. 2019; Cao et al. 2019; Davie et al. 2018). To understand the level of distortion in those cases, we used PCA to embed the hydra data from the original 5000 HVG dimensions into 1-100 PCA dimensions, which is the range typically considered in most scRNA-seq analyses (Luecken and Theis 2019; Andrews et al. 2021). For the entire dataset, the AJD never dropped below 0.85, suggesting that significant distortion is retained by PCA in the range of dimensions typically used for analysis (Fig. 3C). Restriction of the dataset to the single endothelial cell type cluster actually allowed PCA to find significantly lower-distortion embeddings (Fig. 3D), although even in this case the AJD for PCA never

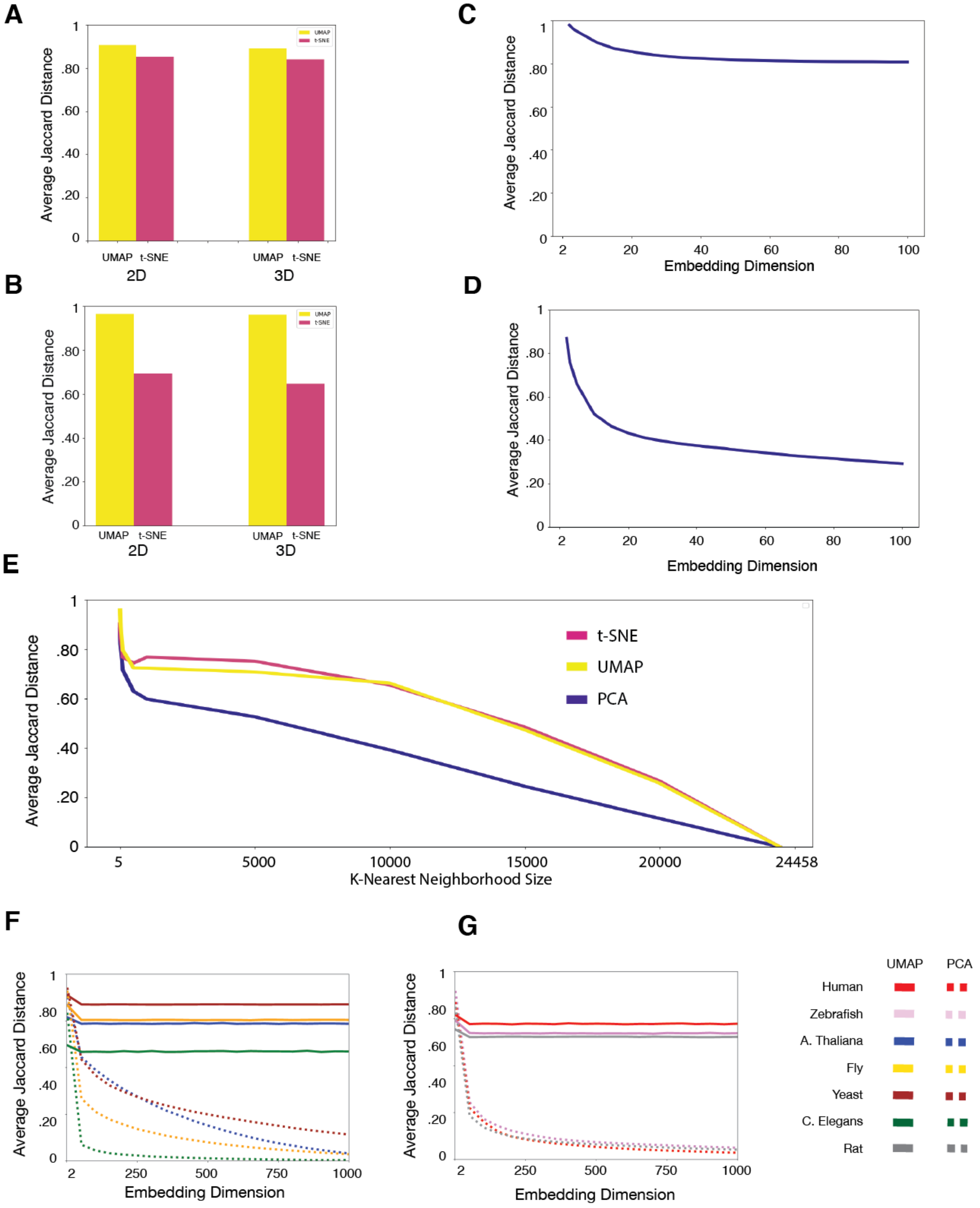


Fig. 3. Distortion introduced by dimensionality reduction in scRNA-seq. (A) The entire dataset from Siebert et al. was subjected to a typical dimensionality reduction pipeline. First, the 5000 most highly varying genes (HVGs) were selected. We then performed PCA down to the "elbow" dimension based on the scree plot, and used both t-SNE and UMAP to project the data down to 2-D. The AJD was then calculated between each of these embeddings and the raw data. **(B)** As in panel A, but for the largest cell-type cluster identified in the Siebert et al. study. A total of 31 dimensions were used for PCA based on the scree plot, and UMAP and t-SNE were

used to generate 2-D projections. Note that t-SNE introduces more distortion for this smaller dataset, while UMAP performs better on this single cluster compared to the entire dataset. **(C)** Here, we used PCA to embed the raw data that was initially reduced to 100- Dimensions from the entire hydra data set into dimensions from 1-100, a typical range used in scRNA-seq studies. Note that the AJD does not drop below 0.8 in this range. **(D)** As in panel C, but for the largest cluster identified by Siebert et al. On this smaller, more focused data set, PCA can find a lower distortion embedding below 100 dimensions, but the distortion is still relatively high at 0.4. **(D)** Average Jaccard Distances vs. neighborhood size (i.e., *k*) for the hydra data set. Here, the data was reduced to 45 dimensions using PCA, and the PCA curve represents the AJD of this PCA embedding as a function of neighborhood size. The PCA embedding was then used to generate 2-D UMAP and t-SNE projections. In all cases, the AJD only approaches 0 when the neighborhood size includes almost the entire dataset, indicating that distortion from dimensionality reduction is highly non-local. **(F)** Average Jaccard Distance vs. Embedding Dimension for a number of recent invertebrate scRNA -Seq studies. The entire data set from each study was used to generate this figure; see refs. (Siebert et al. 2019; Rosenberg et al. 2018; Cao et al. 2019; Davie et al. 2018; Taylor et al. 2019; Zhong et al. 2018; Farrell et al. 2018 ; Jean-Baptiste et al. 2019) for the relevant data sets. The solid line represents the UMAP embedding, and the dashed line represents the PCA embedding. Note that here we used a larger range of embedding dimensions, from 1 to 1000. While PCA can generate low-distortion embeddings for these data sets, it requires more than the typical number of dimensions to do so. Even in higher dimensions, however, UMAP fails to find a low-distortion embedding. **(G)** Average Jaccard Distance vs. Embedding Dimension for UMAP and PCA applied to Vertebrate scRNA-seq studies. As in panel **F**, PCA can generate a low-distortion embedding given enough dimensions, but UMAP cannot.

dropped below 0.35. These findings indicate that, depending on the complexity of the data set in question, using PCA to reduce the dimensionality of the data introduces significant distortion, particularly in the range of dimensions that are most often used for scRNA-seq analysis.

One key parameter of our AJD analysis is the value of *k* used to set the size of the neighborhood that is compared between the original data and the embedding. The results from Figs. 3A-D were all obtained for *k* = 20, which is a relatively local perspective on the neighborhoods involved. This raises the question of whether the distortion introduced by both PCA and NDR methods is simply a permutation of highly local neighborhoods, or whether the distortion is more global in nature. To test this, we considered changing the value of *k* for the hydra data set from relatively local neighborhoods (say, k = 5-20) to values of k that essentially include the entire dataset (Fig. 3E). For this analysis, we generated 2-D embeddings for t-SNE and UMAP (since these techniques are primarily used for visualization) and used 45 dimensions for the PCA embedding based on inspection of the scree plot. As one can see from Fig. 3E, much of the distortion due to dimensionality reduction is *highly non-local*. The AJD remains above 0.75 for both t-SNE and UMAP until the neighborhoods include ~12,000 cells, or over half of the dataset. While the distortion in the PCA embedding is generally lower than that of the t-SNE and UMAP representations, the also AJD remains above 0.5 until the value of k reaches many thousands of cells. Regardless of the technique, the AJD only reaches relatively low values (~0.05) until nearly the entire dataset is included in the “neighborhood” (Fig. 3E). This indicates that the distortion introduced by dimensionality reduction is not purely local, but rather induces large-scale changes in the structure of the data.

The analyses described above focus on just a single data set, and while the range of dimensions considered is typical of scRNA-seq analyses (Luecken and Theis 2019) one could imagine expanding the range of dimensions to ask whether or not dimensionality reduction techniques can effectively embed scRNA-seq data in higher-dimensional spaces. To answer these questions, we collected a representative sample of recent scRNA-seq studies for both invertebrate and vertebrate animals (Siebert et al. 2019; Cao et al. 2019; Davie et al. 2018; Jean-Baptiste et al. 2019; Taylor et al. 2019; Zhong et al. 2018; Farrell et al. 2018 ; Jackson et al. 2019). Many NDR tools are numerically unstable above ~100 dimensions, so for this analysis we focused on UMAP, which performs efficiently across the entire range of dimensions we considered. We compared the AJD of the UMAP embedding with that for PCA across a range of dimensions form 1-1000; above 1000 dimensions, there are too few cells to robustly estimate the principal components for the studies we considered. We found that, as with our results on simulated data, UMAP's performance did not improve significantly as the dimensionality of the embedding increased, and in most cases the AJD never drops below 0.6. In contrast, we found that PCA was able to generate low distortion embeddings with AJD values between 0.05 and 0.1 (i.e. a 5-10% disruption of local neighborhoods) for nearly all of the data sets considered (Figs. 3F and G). Achieving such low-distortion embeddings, however, required many more dimensions than are typically employed for PCA in scRNA-seq studies (between 250 and 750 in most data sets). While these representations are certainly not "low-dimensional," they do represent a significant reduction from the 20,000-40,000 dimensions present in the original datasets (Siebert et al. 2019; Cao et al. 2019; Davie et al. 2018; Zhong et al. 2018; Farrell et al. 2018 ; Jackson et al. 2019; Jean-Baptiste et al. 2019; Taylor et al. 2019).

In order to confirm that the observed distortion wasn't unique to these two studies, we next selected a wide variety of scRNA-seq studies from a diverse set of model organisms, both vertebrate (Fig. 3e) and invertebrate (Fig. 3f) and repeated our analysis in Seurat, using the dimensionality reduction techniques PCA and UMAP (Fig. 3e). In every case, the distortion introduced by UMAP was substantial, and the technique consistently failed to find a low-distortion embedding even in higher dimensions. The performance of PCA varied from data set to data set, but often needed well over 100 dimensions to represent the data with low levels of distortion (e.g. AJD < 0.05). Interestingly, we found that NDR techniques also failed to find low-distortion embeddings for several standard machine learning data sets of considerably smaller size and complexity than scRNA-seq data (Supporting Info).

These results indicate that dimensionality reduction likely introduces significant distortion into data not only reduced to two dimensions, which is commonly used for visualization and some data analysis, but even in higher-dimensional representations of the data. As some degree of dimensionality reduction is an integral part of essentially every scRNA-seq data analysis pipeline, it is clear that the vast majority of scRNA-seq studies are carried out on representations that are likely highly distorted relative to the original neighbor relationships present in the data.

Evaluating the Ability of PCA to “De-noise” Data

Our analysis above assumes that the original neighborhoods present in the data are useful as a “ground truth” for making a comparison between high-dimensional data and lower-dimensional representations. While this makes sense for simulated data, which can be generated without noise, real-world scRNA-seq data is obtained from a noisy experimental technique (Luecken and Theis 2019; Andrews et al. 2021), and thus the original neighborhoods might not represent the true relationships present in the data. It has thus been suggested that PCA might be able to productively “de-noise” this kind of data by finding the principal directions of variance. In other words, the main PCA components should capture the true biological variation within the data; the remaining components can be thought of as capturing the influence of “noise” and can thus be discarded (Lun et al. 2016).

Under this scenario, one might suggest that we should take the PCA embedding as the ground truth for scRNA-seq data, and calculate the AJD relative to that embedding for further downstream steps like visualization or, in some cases, clustering in the 2-D t-SNE or UMAP representation (Rosenberg et al. 2018; Jean-Baptiste et al. 2019; Taylor et al. 2019). In other words, rather than using the set of *k*-NN in the original data as the “true neighbors,” one could argue that the PCA representation of the data should be used as the source of the true set of neighbors for the AJD calculation. It is currently unclear, however, if PCA can indeed remove noise and recover a true set of neighbors in noisy data.

To test this, we generated a number of different synthetic data sets for which we know both the true dimensionality of the data and the original set of true neighbors. We then added noise to this data at increasing levels and used PCA to embed this noisy data. This allowed us to compare the neighborhoods in the PCA embedding to the original neighborhoods using the AJD metric (Fig. 4A). The idea here is that, if the PCA embedding has an AJD of 0 relative to the original data (before noise was added), then PCA is indeed able to remove or reverse the impact of noise. If the AJD is greater than 0, however, this indicates that PCA cannot successfully de-noise the data.

We applied this approach to several types of simulated data sets: hyperspheres, multivariate Gaussian distributions, and simulated scRNA-seq data generated using the PROSSTT package (Papadapolous et al. 2019). Here, we used the PROSST package rather than Splatter to simulate scRNA-seq data with more realistic branching trajectories. The noise added to the data was sampled from a Gaussian distribution with a specified standard deviation and added independently to each simulated "gene" in the data set. To investigate this effect in more realistic settings, we also added noise to the hydra data set (Siebert et al. 2019). Further details on how we generated this noisy data can be found in the Supplementary Information. To evaluate the performance of PCA, we generated the PCA embedding into the correct latent dimension for the synthetic data sets and the standard "elbow" dimension for the hydra data, and then calculated the AJD between the original, noiseless data and the PCA embedding. We found that, once noise increased above a very low level, the AJD increased rapidly to 1 for each data set

(Fig. 4B). In other words, unless noise levels are very small, PCA cannot productively de-noise the data.

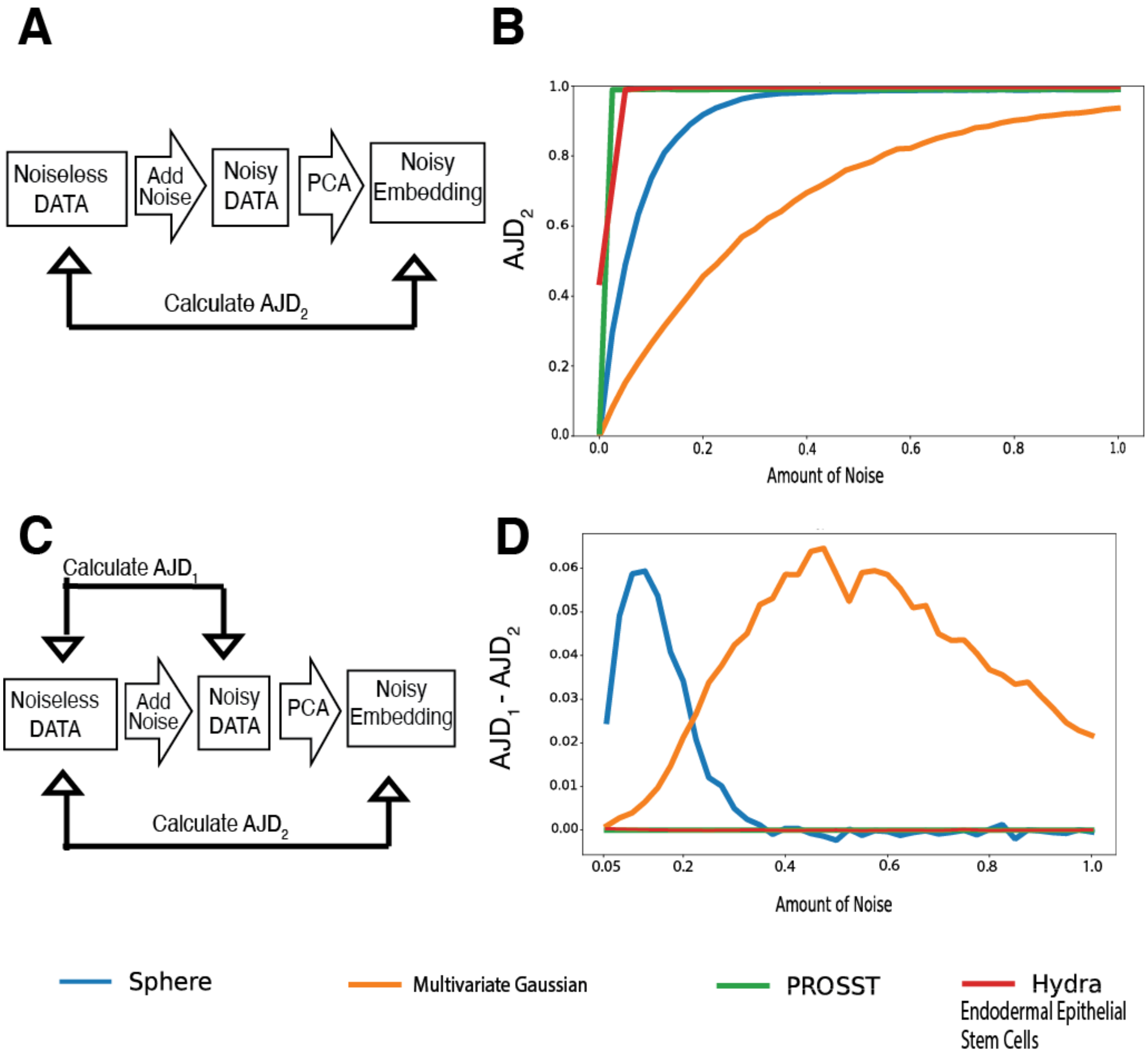


Figure 4. Does PCA Remove Noise? (A) Schematic of the experiment to test whether PCA can recover the original neighborhoods after noise has been introduced into the dataset. In this experiment, Gaussian noise is added to each element in the dataset then PCA is done down the latent dimensionality of the dataset (which is known in the synthetic datasets, and is estimated using the "elbow" in the scree plot for the Hydra endodermal epithelial stem cells) **(B)** Plot showing the AJD of the PCA embedding as a function of the amount of Gaussian noise added to the dataset. In all cases, adding even a small amount of noise resulted in high AJD values, indicating that PCA cannot recover the true local neighborhood structure in its embeddings in the presence of noise. **(C)** Schematic of the experiment whether PCA can improve the neighborhood structure of data with noise and return it to a state with no noise. In this case, the denoising effect is defined by the difference of AJD between the Noiseless Data and the Noisy Data (AJD_1) and

the Noiseless Data and the PCA embedding created after adding Gaussian Noise (AJD_2). The difference between AJD_2 and AJD_1 is the measure of the denoising effect of PCA, as it directly measures how much the AJD has improved with regards to original “ground truth” neighborhoods as a result of using PCA on the noisy data. **(D)** Plot showing the denoising effect with regard to the amount of noise in the Dataset. With the exception of the Multivariate Gaussian, PCA can only denoise the data if the amount of noise added is extremely small. Even then, the improvement in the AJD is extremely limited, less than 6 percent.

One issue with the above analysis is that, when noise is very small, the noise itself might not actually impact the neighborhoods to a significant degree. In other words, imagine that we add so little noise to the data that the noise does not change the neighborhood relationships among the points. In that scenario, it would not be surprising that PCA could recover the original neighborhoods, since the noise itself did not change them. To investigate this, we compared the AJD between the original, noiseless data and the noisy data (which we term AJD_1) and the AJD between the original data and the PCA embedding (AJD_2, Fig. 4C). The difference between these AJDs, $AJD_2 – AJD_1$, represents the improvement in AJD that PCA provides. In other words, this quantifies how much PCA reduces the impact of noise on the structure of local neighborhoods.

We applied this second metric to each of the data sets described above and found that PCA provides very little de-noising capacity (Fig. 4D). For both the hydra data and simulated hyperspheres, PCA was essentially unable to de-noise the data at all. in the case of the PROSTT simulated data and the multivariate Gaussian, PCA was able to provide a small improvement in AJD (up to 0.06) but generally only when noise levels are relatively small.

Taken together, our findings suggest that, despite claims to the contrary in the literature (Wagner et al. 2019), PCA does not generally have the capacity to “de-noise” data in such a way that it recovers the structure of local neighborhoods (Fig. 4). Indeed, PCA can only moderately de-noise data if that noise is both small and orthogonal to the underlying structure within the data (see Supplementary Fig. 8). This finding makes intuitive sense; as an unsupervised technique, PCA has no way to distinguish true variation in the data from noise. An array of statistical studies have demonstrated that, due to the relatively low capture probability of individual mRNA molecules, scRNA-seq experiments have relatively high levels of technical noise, which affects every gene in the data set (Luecken and Theis 2019; Andrews et al. 2021). As such, it is extremely unlikely that scRNA-seq data meets the narrow set of criteria that would allow PCA to restore a set of original neighborhoods that are disrupted by noise. Of course, since there is significant noise in the experiment, the neighborhoods observed in the data may not be the original neighborhoods of the cells in question. In the absence of more reliable experimental techniques, the raw scRNA-seq data is the best representation of the gene expression patterns available. As such, we consider the original set of neighborhoods in the data to be an appropriate "ground truth" for the purposes of evaluating distortion introduced by dimensionality reduction.

Evaluating the Effect of Distortion on Downstream Analyses

As mentioned above, the "standard" pipeline of scRNA-seq data analysis entails both several dimensionality reduction steps and a series of linear and non-linear transformations to the data (Luecken and Theis 2019; Andrews et al. 2021). In our above analysis, we focused on minimally processed scRNA-seq data where the raw counts were just corrected for doublets, batch effects, and other common sources of technical noise in the scRNA-seq experiment. While this allowed us to focus on the impact of PCA and NDR techniques on the local structure of the data, it is unclear how much each of the common steps in scRNA-seq analysis might also influence neighborhood structure. For instance, raw counts are generally first normalized to Counts Per Million (CPM) so that each cell has a total of 1 million counts; this removes variation in the data that comes from different total numbers of counts (i.e. "read depth") between cells (Luecken and Theis 2019; Andrews et al. 2021). The data are then typically subjected to a log

(CPM + 1) transformation. This dataset is then used to identify the subset of “Highly Variable Genes” (HVGs) that display significantly more variability between cells in the experiment than one would expect according to a simple null model. Each of these steps can introduce distortion into the data, even before PCA or NDR techniques are used to reduce the dimensionality further. Analyses like cell type clustering are only performed after all of these transformations, and it is unclear to what extent this entire pipeline affects both the structure of the data and the results of those analyses.

To quantify the impact of these steps on the structure of the data, we first focused on measuring the AJD between the original data and the results of typical analysis pipelines applied to a wide variety of data sets. We used the Seurat package in R (Butler et al. 2018) to perform these analyses, partially because of the popularity of the package and partially because the original analysis of the data was performed using Seurat for the datasets we chose. For each study we used the same embedding dimension for PCA as was used by the original investigators. In Table 1, we report the total level of distortion introduced by this pipeline up to either the PCA step or after UMAP. Clearly, the local structure of the data is significantly disrupted by the pipeline at the PCA step (with AJDs generally above 0.9) and is almost entirely lost downstream of the final NDR step. Note that the UMAP results in Table 1 were obtained using the default settings for the free parameters in the UMAP algorithm, without any attempt at optimization. To see if the results could be improved with parameter optimization, we performed a grid search across values of the two UMAP parameters for the hydra data (Supplementary Material Table 2). While some improvement in distortion is possible, the AJD remained over 0.78 regardless of the parameters, suggesting that parameter optimization cannot fully resolve this issue. We found similar results using AJD to optimize the t-SNE parameters (Supplementary Material Table 3).

These results suggest that the AJD could serve as a useful metric for optimizing the parameters of NDR approaches, but also demonstrate that significant distortion is almost impossible to avoid for 2-D embeddings of scRNA-seq data.

Table 1

Study	Model Organism	Number of PCs	AJD after PCA	AJD after UMAP
(Siebert et al. 2019)	*Hydra vulgaris*	31	0.87	0.92
(Jean-Baptiste et al. 2019)	*Arabidopsis thaliana*	25	0.75	0.81
(Farrell et al. 2018)	*Danio rerio* (Zebrafish)	97	0.90	0.92
(Taylor et al. 2019)	*Caenorhabditis elegans*	125	0.94	0.95
(Davie et al. 2018)	*Drosophila melanogaster* (Fruit Fly)	82	0.94	0.95
(Ma et al. 2019)	*Homo sapiens*	20	0.90	0.91
(Mays et al. 2018)	*Rattus norvegicus*	13	0.99	0.99

Table 1. Average Jaccard distance (AJD) between the minimally processed (raw) scRNA-seq datasets and the representations produced after the "standard" pipeline. Note that the PCA column represents the AJD between the raw data and the data after CPM normalization, log(CPM +1) transformation, HVG identification, and PCA down to the dimension used by the authors of the indicated study. This pipeline is suggested in Seurat and was used by all the authors of the studies in question. As is standard in the field, UMAP was applied to the data after the PCA step to generate the 2-D visualization, and the UMAP column reports the AJD between the visualization and the original data. We used a neighborhood size of 20 for calculating these AJD values.

As mentioned above, one of the most common applications of scRNA-seq analysis is in the identification of distinct cell types in the data, which is almost always done by clustering the cells after application of the "standard pipeline" described above. Our findings from Table 1 suggest that the clusters obtained after all of this dimensionality reduction might be quite distinct from the clusters one would identify in the original data. To test this, we performed

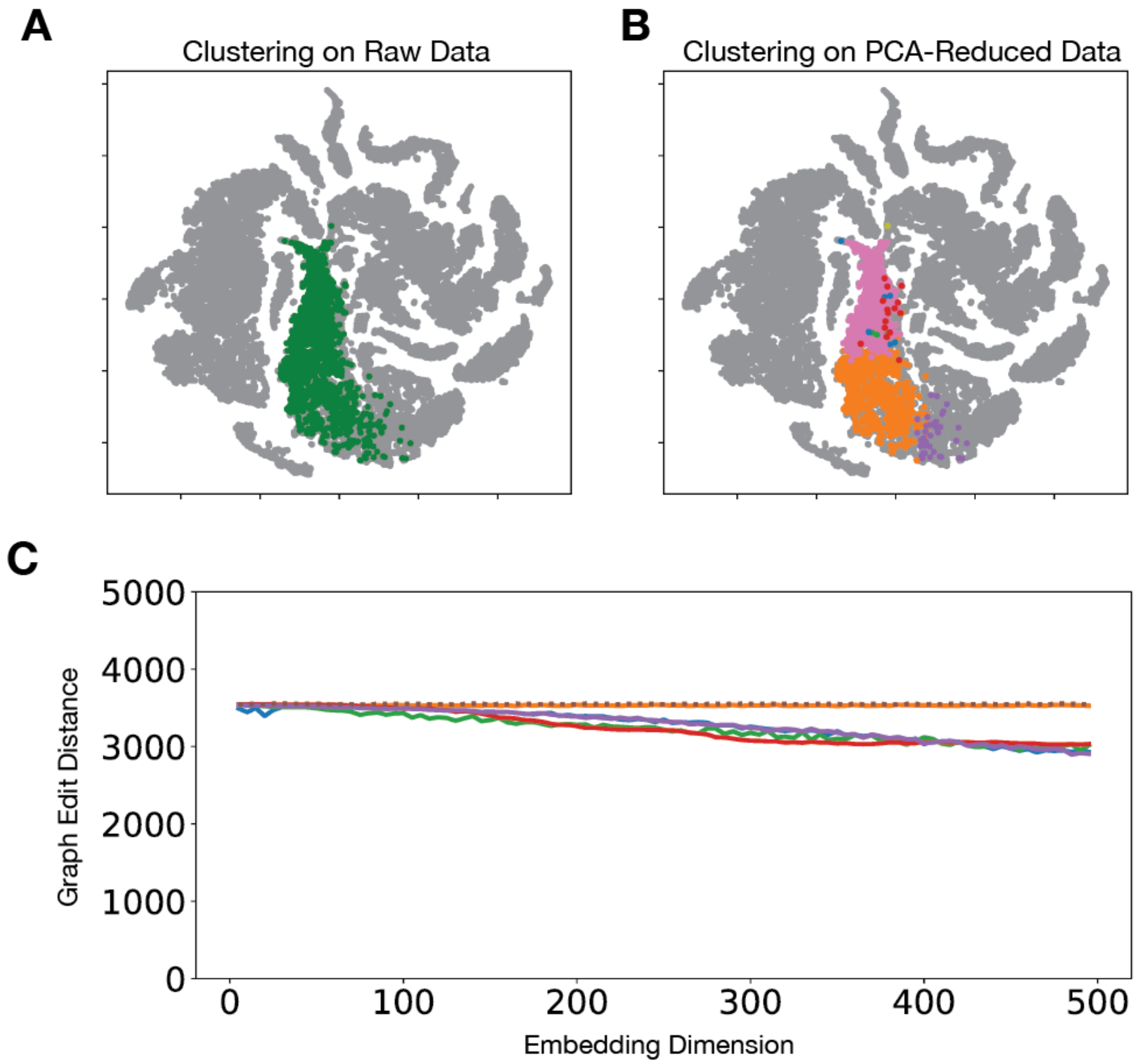


Fig. 5. Distortion and its influence on downstream analyses. (A) Distortion vs. neighborhood size. A single cell RNA sequencing dataset is filtered for highly varying genes. The data is then embedded into a 45-dimensional space using PCA. (The choice of 45 principal components was based on inspection of a scree plot) The data is then embedded into 2 dimensions using t-SNE and UMAP. Average Jaccard Distances are calculated between the raw data and the PCA embedding, as well as between the raw data and the 2-dimensional embeddings using various values for the k-nearest neighbor search. (B) The result of clustering of scRNA-seq data in the original, ambient dimension (left), and the result using the same clustering algorithm with the same parameters on PCA-reduced representation of the data. Only a subset of the points is colored for clarity. The graphs were produced using t-SNE for the purpose of visualization only, as the t-SNE embedding loses much of the structure of the data. (C) The Graph Edit Distance between a minimum spanning tree constructed in the ambient

space and a minimum spanning tree constructed in the NDR-reduced representation. The dotted line corresponds to a random embedding that retains none of the original information.

clustering on the entire Hydra data set, first on the raw data, and then after application of the standard pipeline up to the PCA step using the number of components employed by the authors in their original work (Siebert et al. 2019). We used the standard Louvain clustering algorithm with the default parameters in Seurat (see Methods) (Butler et al. 2018). To visualize the impact of the standard pipeline on clustering results, we chose the largest cluster we obtained from clustering on the raw data and colored those points green on a t-SNE visualization (Fig. 5A). We then colored those same cells according to the clusters obtained from the standard pipeline data (Fig. 5B). Although this t-SNE is used simply for visualization purposes, given the large amount of distortion it introduces, visual inspection of these results clearly indicates that the resulting clusters are very different.

While these results suggest that cell type clustering may be heavily influenced by dimensionality reduction, a visualization like this is difficult to interpret quantitatively. We thus used the Adjusted Rand Index (ARI), a measurement of similarity in clustering results, to quantify the similarity of the clusters obtained from either the PCA or UMAP step of the standard pipeline with those obtained from clustering on the raw data (Table 2). Because clustering only makes sense in the case where there are multiple distinct cell types, we applied this analysis only to those studies where it was computationally feasible to analyze all cells in the data set. As in Fig. 5, we obtained clusters using the standard procedure in Seurat (see Methods).

We found that the ARI values between the clusters obtained from raw data and the clusters based on the PCA-reduced data indicates significant differences between the clusters in every case. Clustering in the 2-D UMAP space results in even more divergence between the

Table 2

Study	**Model Organism**	**ARI: PCA**	**ARI: UMAP**
(Siebert et al. 2019)	*Hydra vulgaris*	0.61	0.43
(Jean-Baptiste et al. 2019)	*Arabidopsis thaliana*	0.53	0.45
(Jackson et al. 2019)	*Saccharomyces cerevisiae* (Yeast)	0.25	0.14
(Siebert et al. 2019)	*Danio rerio* (Zebrafish)	0.12	0.09
(Taylor et al. 2019)	*Caenorhabditis elegans* (Worm)	0.31	0.23
(Ma et al. 2019)	*Homo sapiens* (Human)	0.36	0.21
(Davie et al. 2018)	*Drosophila melanogaster* (Fruit Fly)	0.27	0.12

Table 2. Adjusted Rand Index (ARI) between clustering performed on the minimally processed (raw) scRNA-seq datasets and clustering performed on representations produced by dimensionality reduction. In each case, the number of PCs used for PCA is the same as in the original study, and UMAP into 2 dimensions is performed downstream of PCA. In every case, the clustering is substantially different after PCA, and even more dissimilar after UMAP.

clusters obtained, with ARI values close to 0 in several cases (Table 2). This indicates that the overlap between clustering in UMAP space vs. clustering in the raw space is roughly equivalent to what one would expect if the two different clusterings were generated completely at random. Overall, these results suggest that distortion introduced by both linear and non-linear dimensionality reduction can significantly change the classification of cells into specific cell types based on clustering in scRNA-seq data.

Pseudotime ordering attempts to use cells captured at various points along a differentiation or developmental trajectory to infer the underlying trajectory itself (Trapnell et al. 2014b). A large number of algorithms have been proposed for this analysis, but perhaps the most classic approach involves the calculation of a *minimum spanning tree* that connects the beginning and end point in the trajectory (Trapnell et al. 2014b). This tree is formed by linking cells in close proximity to each other to form a graph, typically after NDR is performed. Because NDR readily changes both the local and global relationships between cells in the data set (Figs. 2 and 3), we hypothesized that the trees produced by analyzing data after NDR would not closely resemble trees formed using the original data. To test this, we calculated the graph edit distance between trees formed from the raw data and after various NDR techniques were used to project the data into a variety of different dimensions (Fig. 5C). For comparison, we also generated a random embedding by simply assigning each cell to a random point in the reduced-dimensional space (see Methods). The graph edit distances obtained from the NDR techniques and from the random embedding are similar until embedding dimensions of ~100 are reached (Fig. 5C). Even above 100 dimensions, the improvement in the graph edit distance relative to a random embedding is not very large. Because pseudotime trees are usually built using 2- or 3-dimensional representations based on t-SNE, UMAP or similar techniques (Trapnell et al. 2014b; Saelens et al. 2019), our findings suggest that distortion caused by NDR could have a large effect on the results. Even pseudotime inferences techniques that do not form minimum spanning trees are based on analysis of scRNA-seq data after significant dimensionality reduction, suggesting that distortion has a wide-ranging impact on this type of analysis (Saelens et al. 2019).

Discussion

The capacity to generate high-dimensional data is currently in the process of revolutionizing scientific inquiry. scRNA-seq, for example, has the potential to drive significant advances in our understanding of the evolution and differentiation of cell types, the progression of cellular state during development and disease, and a host of other critical biological phenomena (Luecken and Theis 2019; Andrews et al. 2021). Yet the very thing that makes this technique so powerful – the ability to simultaneously measure the expression level of tens of thousands of genes within a single cell – also entails the curse of dimensionality and thus complicates the analyses needed to extract meaning from the data. As such, dimensionality reduction has become an indispensable part of scRNA-seq data analysis (Moon et al. 2018; Luecken and Theis 2019; Andrews et al. 2021). It is currently unclear, however, to what extent dimensionality reduction disrupts the underlying structure of the data itself.

Distortion from dimensionality reduction can take several forms. Much of the previous work on this problem has focused on the extent to which the process changes the distances between points (McInnes et al. 2018 ; Laurens van der Maaten and Geoffrey E. 2008). Our work highlights that there are even larger problems with dimensionality reduction than just distortion of distances. For one, even in possession of a perfect technique, one cannot reduce the dimensionality of the data to arbitrarily low dimensions without creating large numbers of discontinuities in local neighborhoods and other distortions in the data. In the case of points taken from the surface of a 3-D sphere, for instance, it is mathematically impossible to project those points into a 2-D representation without introducing discontinuities in local neighborhoods into the data (e.g., the scattering of the rainbow pattern in Fig. 1c). Many analyses commonly performed with scRNA-seq data, including cell type clustering, RNA velocity, and pseudotime ordering, rely at least in part on the local relationships between data points (Trapnell et al. 2014b; Luecken and Theis 2019; Andrews et al. 2021; La Manno et al. 2018). The introduction of discontinuities thus has the potential to significantly impact the results of a wide range of downstream analyses.

A second problem is the fact that, even if it is theoretically possible to represent the data in a given dimension, available techniques may not be capable of finding that representation. Unfortunately, it is currently impossible to evaluate the extent to which either of these issues have an impact on the analysis of scRNA-seq data (or, indeed, any high-dimensionality data).

Here, we developed a straightforward metric that quantifies the extent to which discontinuities of the type exemplified in Fig. 1C would impact the analysis of any given data set.

One immediate application of this metric is in the discovery of the appropriate latent dimension of a given data set. In testing this use case on data sampled from hyperspheres, however, we found that all NDR techniques currently in widespread use are far from perfect (Fig. 2). Indeed, none of the techniques we tested could find a true embedding for even a 20-dimensional hypersphere, despite a complete lack of noise in the data and the fact that the embedding in this case was rather trivial (and known *a priori*). We found that this problem was not limited only to hyperspheres, but to also to generic multivariate Gaussians and simulated scRNA-seq data generated using the Splatter algorithm (Zappia et al. 2017) (Fig. 2). This finding suggests that fundamental work is needed to develop new and more effective NDR techniques. We expect that both the AJD metric we developed and the simulated data sets that we explored will prove useful in the design and testing of these algorithms.

Application of our metric to scRNA-seq data revealed that the problem there is even worse than for hyperspheres (Fig. 3). For instance, it is currently common to use t-SNE or UMAP to reduce scRNA-seq data to two dimensions for visualizations and, in many cases, downstream data analysis (Trapnell et al. 2014b; Rosenberg et al. 2018; Jean-Baptiste et al. 2019; Taylor et al. 2019). Our work revealed that nearly 100% of the local neighborhood structure is disrupted by this kind of dimensionality reduction. We found that this level of distortion has a significant effect on the results of common analyses such as cell type clustering and pseudotime ordering (Tables 1 and 2 and Fig. 5). Interestingly, we also found that PCA, which is often thought to "de-noise" the data, is extremely unlikely to recover a set of true neighborhood relationships given the high levels of noise typically observed in scRNA-seq experiments (Eraslan et al. 2019; Kim et al. 2015; Townes et al. 2019) (Fig. 4).

There are several practical implications of our findings for routine scRNA-seq analysis. A straightforward recommendation flowing from this work is to exercise caution when analyzing data in dimensions that are significantly smaller than the ambient space of the original measurements, particularly the 2-D representations generated by t-SNE or UMAP. We recommend that practitioners use the AJD to track the distortion they introduce into their data when employing dimensionality reduction and report it so that others can understand potential biases and errors that may affect the results of analyses that rely on local relationships between cells in the dataset. Secondly, the AJD could be used as a parameter to optimize several steps in the analysis pipeline, from choosing the appropriate PCA dimension (i.e., an alternative to the scree plot, Fig. 3) to optimizing the parameters of NDR techniques (Tables 1, 2 and 3 Supplementary Material).

Our findings, and the recommendations above, might at first glance seem to conflict with the fact that most scRNA-seq studies ultimately produce results that are broadly consistent with orthogonal data regarding the system under study. For instance, t-SNE and UMAP plots still tend to place cells of similar type close to one another. This is often checked by coloring cells according to the expression of marker genes that are known to be associated with certain cell types, and finding that those cells tend to cluster together, at least on visual inspection (Siebert et al. 2019; Cao et al. 2019; Rosenberg et al. 2018). Similarly, pseudotime analysis often results in expression dynamics that broadly correlate with known expression dynamics obtained from other techniques (Jean-Baptiste et al. 2019; Zhong et al. 2018; Bach et al. 2017). While this agreement seems reassuring, there is a subtle issue with this kind of analysis.

Each of the dimensionality reduction techniques mentioned above are governed by one or more parameters. A small adjustment in any of these parameters can result in vastly different representations of the data (Supplementary Fig. 6). How does one decide the appropriate values for the parameters? In practice, one first selects marker genes that they know correspond to certain cell types based on previous studies. The expectation in this case is that the analysis pipeline, which entails several steps of dimensionality reduction, will have been executed correctly when the marker genes cluster more-or-less according to prior knowledge. Adjusting the parameters of the algorithm until agreement is achieved, the researcher concludes that these are the correct parameter values, and this is the correct representation because the result has been "validated" by prior knowledge. Other observed clusters can then be interpreted as representing new cell types. Popular packages, such as Seurat, include suggestions along these lines for users in their documentation, particularly when looking for rare cell types in a population (Butler et al. 2018).

The problem with this approach is that it is inherently biased to reproduce known aspects of the system in question. To see why, suppose that the biological ground truth doesn't agree with prior biological knowledge. The researcher will discard such a result and adjust the parameters of the analysis pipeline until the representation comes into agreement with their expectations. In other words, if prior knowledge is used to guide the analysis, the fact that one ultimately sees agreement between the result and that prior knowledge is no guarantee that the analysis itself is sound. This is true even if the marker genes used to guide clustering or other analysis are different from the ones used for "validation," since it is unlikely that any such sets of genes will be truly independent of one another. Thus, while many scRNA-seq analyses agree with well-established prior knowledge, that in no way guarantees that distortion due to dimensionality reduction has not significantly impacted the analysis.

Of course, one question raised by our results is whether or not meaningful dimensionality reduction of scRNA-seq data is possible at all. The poor performance of NDR techniques on the simple hypersphere tests makes it difficult to say whether the results we obtained for scRNA-seq data are due to the limitations of available techniques or because the data do not actually lie on a low-dimensional manifold. We note, however, that NDR techniques failed to find meaningful embeddings even for standard non-scRNA-seq data sets used in machine learning research (Supporting Info), strongly suggesting that the issue here lies with the techniques themselves, rather than representing limitations of the individual data sets. The only technique that we found to provide something close to a "true" embedding, PCA, does so only at dimensionalities that are much larger than those typically used. Indeed, PCA sometimes only finds a true embedding at the largest possible dimension that can be obtained by the technique (Fig. 3). The development of new NDR techniques that are more effective at finding true embeddings thus represent a critical step in answering central questions not only in cell biology, but across all scientific disciplines that rely on the analysis of high-dimensional data. Until such techniques are developed, the relentless expansion of single-cell genomics to larger and larger scales may provide a wealth of new data that cannot be optimally mined for its biological insights.

Acknowledgments

We thank members of the Deeds and Ray labs, Tom Kolokotrones, Alan Garfinkel, and Lukas M. Weber (University of Zurich, Switzerland) for useful discussions. This work was supported in part by an Institutional Development Award (IDeA) from the National Institute of General Medical Sciences of the National Institutes of Health under Award Number P20

GM103638 to JCJR, and a grant from the National Institute of General Medical Sciences of the National Institutes of Health, Award Number R01 GM143378, to EJD. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

Methods

Average Jaccard Distance

For each data point, the neighborhood consisting of the nearest k-neighbors were found in the ambient space, call this set A, and the NDR-reduced space, call this set B, using sklearn.neighbors.NearestNeighbors. We employed the ball-tree algorithm in both cases. To calculate the Jaccard distance between A and B, we used the usual definition:

$$D_J(A,B) = \frac{|A \cup B| - |A \cap B|}{|A \cup B|}$$

The Average Jaccard Distance was calculated by taking the arithmetic mean of the Jaccard distance for every point.

Sampling of Hyperspheres

To create a synthetic dataset consisting of m uniformly distributed samples in an n-dimensional spherical manifold in d-dimensional space, we used the following method: For each of the m data points, we sampled from a standard normal distribution n times (using the Python Numpy method numpy.random.normal(0,1)). This method ensured that the sampling on the sphere was uniform. These samples became the first n coordinates of a vector. The remaining n+1 to d coordinates were filled with zeros. We then normalized each vector to length 1.

Dimensionality Reduction

We executed dimensionality reduction with t-SNE, Isomap, PCA, Spectral Embedding, Multidimensional Scaling, LLE, and LTSA using the implementations in Scikit-learn (Pedregosa et al. 2011). For the methods UMAP and diffusion maps, we used umap-learn (McInnes et al. 2018) and pydiffmap (Berry and Harlim 2016), respectively. We implemented PCA using sklearn.decomposition.PCA. We used default parameters except where otherwise noted.

scRNA-seq Data

The study from Siebert et al. is published on the Broad Institute's single cell portal: https://portals.broadinstitute.org/single_cell/study/SCP260/stem-cell-differentiation-trajectories-in-hydra-resolved-at-single-cell-resolution.

The study from Cao et al. is published on The Gene Expression Omnibus:

https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE119945

The .txt files were converted to .csv files corresponding to individual clusters, and the data were loaded into Python pandas (https://pandas.pydata.org/) dataframes for dimensionality reduction.

Minimum Spanning Tree and Graph Edit Distance

The minimum spanning tree in the ambient space, mst$_1$, and the minimum spanning tree in the NDR-reduced space, mst$_2$, were constructed using the Python function scipy.sparse.csgraph.minimum_spanning_tree. The graph edit distance was calculated in Python according to the following equation:

$$GED(mst_1, mst_2) = \min_{\{e_1,\ldots,e_k\}\in P(mst_1,mst_2)} \sum_{i=1}^{k} c(e_i)$$

Where $P(mst_1, mst_2)$ is the set of edit paths transforming mst$_1$ into mst$_2$ and $c(e_i)$ is the cost of each graph edit operation e_i. The cost of deleting a vertex and the cost of adding a vertex were both weighted as 1.

As a control, a random embedding was created by sampling coordinates from a uniform distribution between -1 and 1. The minimum spanning tree was then computed on this random embedding and the Graph Edit Distance was calculated between this tree and the minimum spanning tree constructed in the ambient space.

Adjusted Rand Index

The Rand index quantifies the similarity between clusters in two partitions U and V (say, cell clusters in the ambient dimension and in a reduced dimension) through a contingency table that classifies pairs of points into four cases: pairs in the same cluster in both partitions (a), pairs in the same cluster in U but not V (b), pairs in the same cluster in V but not U (c), or pairs in different clusters in both partitions (d). It takes a value between 0 and 1. The adjusted Rand index corrects the value by accounting for coincidental/chance clustering and avoiding the tendency of the unadjusted Rand index to approach 1 as the number of clusters increases. It is given by

$ARI = \frac{(n\ 2\ )(a+d)-[(a+b)(a+c)+(c+d)(b+d)]}{(n\ 2\ )^2-[(a+b)(a+c)+(c+d)(b+d)]}$ where n is the number of points and $(n\ 2\ )$ is the total number of possible point pair combinations (Santos and Embrechts 2009).

Replicating scRNA-seq Workflows

To replicate a typical workflow, we used Seurat (Butler et al. 2018) in R. To isolate highly variable genes, we used the data from the function FindVariableFeatures() in Seurat with default parameters. For PCA reduction, we used the ElbowPlot function, with the "elbow" observed to be at 12 PCs.

Our clustering was done in Seurat using the function FindNeighbors() on the specified dimensional space to compute the Shared Nearest Neighbor Graph, followed by the

FindClusters() function. We set the resolution at 0.8, number of random starts at 10, random seed at 0, maximum number of iterations at 10 and we used the standard modularity function.

Evaluating PCA's denoising ability.

To test whether PCA can effectively denoise data. we decided to use 3 synthetic datasets and 1 real sc-RNA-seq datasets: 1000 points uniformly sampled from a 20 Dimensional Hypersphere embedded in 100-Dimensional Space; 1000 points sampled from a 20-Dimensional Multivariate Gaussian; 1000 Points from a tree-lineage structure generated by the Python Package PROSSTT, with 20 genes and 4 branch points; and the Endodermal Epithelial Stem Cell Cluster in the previously used Hydra Dataset.

To each of these datasets, we added Gaussian Noise, both *On-Manifold (*referring to noise being added to the feature columns that were used to define the structure of the manifold) and *Off-Manifold* (referring to noise being added to columns of zeros appended to the end of the dataset to make the full space). For the Hypersphere and Multivariate Gaussian datasets, both the On-Manifold and Off-Manifold Noise was simulated by adding a vector sampled from a Gaussian distribution whose covariance matrix was the identity matrix. For the PROSSTT dataset, the On-manifold noise was simulated by adding a vector sampled from a Gaussian distribution with no covariance and whose variance was proportional to the variance of the aligned feature. The Off-manifold noise was simulated by adding a vector sampled from a Gaussian distribution which had no covariance and whose variance exponentially decreased from the maximum variance observed in the PROSSTT data without noise to the minimum variance observed in the PROSSTT data without noise. For the Hydra, noise was added by adding a vector sampled from a Gaussian distribution no covariance and whose variance was proportional to the variance of the aligned feature. In each case, after noise was added the average Jaccard Distance for each dataset before and after noise was added was calculated (AJD_1).

After noise was added, PCA was done to denoise the data and find the latent dimensionality. For the synthetic datasets generated, the latent dimensionality was known *a priori*. For the Hydra dataset, the latent dimensionality was estimated using the "elbow" of the scree plot of explained variance. To automate the determination of this "elbow" and reduce operator bias introduced to the experiment, the kneelocator function within the Python package Kneed was used, with the sensitivity set to 1.0, the curve parameter set to "convex" and the direction set to "decreasing" to estimate the latent dimensionality of the dataset that PCA would reduce the data to during the denoising process. After PCA was applied to each of the datasets, the Average Jaccard Distance between the High Dimensional Datasets before noise was added and the datasets after PCA was applied *(AJD_2).*

References

Andrews,T.S. *et al.* (2021) Tutorial: guidelines for the computational analysis of single-cell RNA sequencing data. *Nat Protoc*, **16**, 1–9.

Bach,K. *et al.* (2017) Differentiation dynamics of mammary epithelial cells revealed by single-cell RNA sequencing. *Nat Commun*, **8**, 2128.

Berry,T. and Harlim,J. (2016) Variable bandwidth diffusion kernels. *Applied and Computational Harmonic Analysis*, **40**, 68–96.

Butler,A. *et al.* (2018) Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nat Biotechnol*, **36**, 411–420.

Cao,J. *et al.* (2019) The single-cell transcriptional landscape of mammalian organogenesis. *Nature*, **566**, 496–502.

Chalupka,K. *et al.* (2016) Unsupervised Discovery of El Nino Using Causal Feature Learning on Microlevel Climate Data. *arXiv:1605.09370 [physics, stat]*.

Cichocki,A. and Phan,A. (2008) Fast Local Algorithms for Large Scale Nonnegative Matrix and Tensor Factorizations.

Davie,K. *et al.* (2018) A Single-Cell Transcriptome Atlas of the Aging Drosophila Brain. *Cell*, **174**, 982-998.e20.

DeMers,D. and Cottrell,G. (1993) Non-Linear Dimensionality Reduction. In, Hanson,S. *et al.* (eds), *Advances in Neural Information Processing Systems*. Morgan-Kaufmann.

Eraslan,G. *et al.* (2019) Single-cell RNA-seq denoising using a deep count autoencoder. *Nat Commun*, **10**, 390.

Farrell,J.A. *et al.* (2018) Single-cell reconstruction of developmental trajectories during zebrafish embryogenesis. *Science*.

Friedman,J.H. (1997) On Bias, Variance, 0/1—Loss, and the Curse-of-Dimensionality. *Data Mining and Knowledge Discovery*, **1**, 55–77.

Horrocks,T. *et al.* (2019) Geochemical characterisation of rock hydration processes using t-SNE. *Computers & Geosciences*, **124**, 46–57.

Hotelling,H. (1933) Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, **24**, 417–441.

Indyk,P. and Motwani,R. (1998) Approximate nearest neighbors: towards removing the curse of dimensionality. In, *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, STOC '98. Association for Computing Machinery, New York, NY, USA, pp. 604–613.

Jackson,C.A. *et al.* (2019) Gene regulatory network reconstruction using single-cell RNA sequencing of barcoded genotypes in diverse environments.

Jean-Baptiste,K. *et al.* (2019) Dynamics of Gene Expression in Single Root Cells of Arabidopsis thaliana. *Plant Cell*, **31**, 993–1011.

Kim,J.K. *et al.* (2015) Characterizing noise structure in single-cell RNA-seq distinguishes genuine from technical stochastic allelic expression. *Nat Commun*, **6**, 8687.

Knyazev,A.V. (1998) PRECONDITIONED EIGENSOLVERS—AN OXYMORON? *ETNA*, **7**, 1.

Kruskal,J.B. (1964) Nonmetric multidimensional scaling: A numerical method. *Psychometrika*, **29**, 115–129.

La Manno,G. *et al.* (2018) RNA velocity of single cells. *Nature*, **560**, 494–498.

Lake,B.B. *et al.* (2018) Integrative single-cell analysis of transcriptional and epigenetic states in the human adult brain. *Nat Biotechnol*, **36**, 70–80.

Lemmon,E.M. and Lemmon,A.R. (2013) High-Throughput Genomic Data in Systematics and Phylogenetics. *Annual Review of Ecology, Evolution, and Systematics*, **44**, 99–121.

Levandowsky,M. and Winter,D. (1971) Distance between Sets. *Nature*, **234**, 34–35.

Lueken,M.D. and Theis,F.J. (2019) Current best practices in single-cell RNA-seq analysis: a tutorial. *Molecular Systems Biology*, **15**, e8746.

Lun,A.T.L. *et al.* (2016) A step-by-step workflow for low-level analysis of single-cell RNA-seq data with Bioconductor. *F1000Res*, **5**, 2122.

Ma,L. *et al.* (2019) Tumor Cell Biodiversity Drives Microenvironmental Reprogramming in Liver Cancer. *Cancer Cell*, **36**, 418-430.e6.

Maaten,L. van der and Hinton,G. (2008) Visualizing Data using t-SNE. *Journal of Machine Learning Research*, **9**, 2579–2605.

Mays,J.C. *et al.* (2018) Single-cell RNA sequencing of the mammalian pineal gland identifies two pinealocyte subtypes and cell type-specific daily patterns of gene expression. *PLOS ONE*, **13**, e0205883.

McInnes,L. *et al.* (2018) UMAP: Uniform Manifold Approximation and Projection. *Journal of Open Source Software*, **3**, 861.

Moon,K.R. *et al.* (2018) Manifold learning-based methods for analyzing single-cell RNA-sequencing data. *Current Opinion in Systems Biology*, **7**, 36–46.

Ozsolak,F. and Milos,P.M. (2011) RNA sequencing: advances, challenges and opportunities. *Nat Rev Genet*, **12**, 87–98.

Papadopoulos,N. *et al.* (2019) PROSSTT: probabilistic simulation of single-cell RNA-seq data for complex differentiation processes. *Bioinformatics*, **35**, 3517–3519.

Pearson,K. (1901) LIII. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, **2**, 559–572.

Pedregosa,F. *et al.* (2011) Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, **12**, 2825–2830.

Rosenberg,A.B. *et al.* (2018) Single-cell profiling of the developing mouse brain and spinal cord with split-pool barcoding. *Science*.

Roweis,S.T. and Saul,L.K. (2000) Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science*.

Saelens,W. *et al.* (2019) A comparison of single-cell trajectory inference methods. *Nat Biotechnol*, **37**, 547–554.

Santos,J.M. and Embrechts,M. (2009) On the Use of the Adjusted Rand Index as a Metric for Evaluating Supervised Classification. In, Alippi,C. *et al.* (eds), *Artificial Neural Networks – ICANN 2009*, Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, pp. 175–184.

Siebert,S. *et al.* (2019) Stem cell differentiation trajectories in Hydra resolved at single-cell resolution. *Science*.

Stegle,O. *et al.* (2015) Computational and analytical challenges in single-cell transcriptomics. *Nat Rev Genet*, **16**, 133–145.

Taylor,S.R. *et al.* (2021) Molecular topography of an entire nervous system. *Cell*, **184**, 4329-4347.e23.

Tenenbaum,J.B. *et al.* (2000) A Global Geometric Framework for Nonlinear Dimensionality

Reduction. *Science*.

Townes,F.W. *et al.* (2019) Feature selection and dimension reduction for single-cell RNA-Seq based on a multinomial model. *Genome Biology*, **20**, 295.

Trapnell,C. *et al.* (2014) The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nat Biotechnol*, **32**, 381–386.

Wagner,F. *et al.* (2019) Accurate denoising of single-cell RNA-Seq data using unbiased principal component analysis.

Wolf,F.A. *et al.* (2018) SCANPY: large-scale single-cell gene expression data analysis. *Genome Biology*, **19**, 15.

Zappia,L. *et al.* (2017) Splatter: simulation of single-cell RNA sequencing data. *Genome Biology*, **18**, 174.

Zhang,Z. and Zha,H. (2002) Principal Manifolds and Nonlinear Dimension Reduction via Local Tangent Space Alignment. *arXiv:cs/0212008*.

Zhong,S. *et al.* (2018) A single-cell RNA-seq survey of the developmental landscape of the human prefrontal cortex. *Nature*, **555**, 524–528.