Manifold Learning, Warping and Alignment as Tools for Highly Dimensional Data

Alexander G. Hasson[1,2] and Kevin Burrage[1,2]

[1] School of Mathematical Sciences, Queensland University of Technology, Australia
[2] ARC Centre of Excellence for Plant Success in Nature and Agriculture
{hasson,kevin.burrage}@qut.edu.au

Abstract. With highly dimensional data comes significant computational cost in analysis. Traditional dimensionality reductions methods such as Principle Component Analysis are subject to artefacts such as the Guttman effect which make downstream analysis prone to errors. This is due to the inability of linear methods to capture the structure of inherently non-linear relationships within datasets. Recent methods in manifold-learning, manifold-warping and manifold-alignment (ManiLWA) provide the necessary tools to analyse highly dimensional data whilst reducing or eliminating the artefacts found with linear-based approaches.

Keywords: Dimensionality reduction, Manifolds, Manifold Warping

1 Introduction

1.1 Highly Dimensional Data

Since the early 2000s, there has been a necessity to analyse high dimensional datasets. Although sources for datasets have changed, to include data from emerging scientific areas such as metagenomics, the burden or often 'the curse' of dimensionality remains.

Traditionally, statistics and data analysis have relied on many observations of fewer key variables. This is in contrast to the current trend of data to be collections of observations with "voracious, automatic, systematic collection of hyper-informative detail about each observed instance."[32]. As there are now far fewer observations of evergrowing features, traditional statistics and data analysis methods fail[48]. This has lead to two very intertwined methods for when there is significant imbalance between observations and features[42]. New analytical methods specifically designed for high-dimensional data are being developed, as well as research into dimensionality reduction (through feature extraction[58] and feature reduction[82,6]).

1.2 Dimensionality Reduction

Modern dimensionality reduction methods fit into two main classes: supervised and unsupervised methods. In addition, these methods are often designed for

either feature *extraction* or feature *reduction*. Feature extraction transforms the underlying data into a new space determined by new features, whereas feature reduction retains a subset of the original features. For example, two commonly used feature extraction methods are Principle Component Analysis[47] and Linear Discriminant Analysis[26]; unsupervised and supervised methods, respectively. This paper covers mainly unsupervised feature *extraction* methods.

Dimensionality reduction provides several benefits. In terms of computation, working in lower dimensions often improves both time and space complexity. In addition, if the resultant embedding is in very low dimensional space (say, less than or equal to three), then the embedding can be easily visualised[17]. There are numerous other reasons to perform dimensionality reduction, such as to remove noise, but these are often dependent on the method utilised[10].

1.3 Manifolds, the Manifold Assumption and Manifold Learning

Manifolds are smooth topological spaces and are often used to describe higher-dimension locality preserving spaces[78,76]. The 'manifold assumption' is an important assumption in data science that states the data of interest lies on a sub-manifold embedded in much higher dimensional space. That is, for some mapping function f :

$$\text{with } \{X_1, X_2, \dots X_n\} \in \mathbb{R}^D, f : \mathbb{R}^D \to \mathbb{R}^d \text{ exists. Where } d < D.$$

Manifold learning is the concept and practice of determining from a given data set or topological space, the function for which a lower dimensional representation can be realised[36,78]. To determine the effectiveness of the manifold representation, and the lower dimensional embedding, there is need for a measure of 'correctness' often given as 'closeness'[39,50,23]. 'Closeness' refers to points on the original higher dimensional space having their spatial or geometric relationships preserved; points that are close stay close, points that are far away remain so. Measurements are often done along the curvature of the higher-dimensional space, utilising geodesic paths, geometric distances or similar[56].

Reducing the dimensionality of manifolds differs from traditional Euclidean spaces, as the distance along the space differ from the Euclidean distances[79]. Therefore, methods such as Principal Component Analysis (PCA) often create subspaces not faithful to the original topology[47]. This important difference has led to the development of methods for dimensionality reduction, specifically for manifolds[88,68,61].

1.4 k Nearest Neighbour Graphs

k Nearest Neighbours (kNN) is a method that relies on the assumption that data points similar to each other - for example data points within the same class - are close to each other spatially[91]. The method is often used for regression and classification, in the case of the latter the k-nearest points (based on some metric, such as Euclidean distance) is utilised in a consensus-based approach

to predict the class of unknown data points[29]. The parameter k, $k \in \mathbb{Z}^+$, determines how many points to consider in the decision process, and is often iteratively determined via an optimisation process, such as error minimisation in a classification problem[36].

Graphs (often embeddings) can be constructed through the use of the kNN algorithm. For example, each node can be connected to the k nearest nodes. These graphs attempt to represent the 'connectivity' or topology of the dataset[51]. There are several adaptions of these methods, and it appears that lower values of k can preserve local connectivity, with larger values providing global information about a dataset[27]. The resultant graphs are a useful method for manifold learning, and have been used to learn the underlying topology of datasets in a variety of contexts[25].

kNN can be used as a dimensionality reduction method, largely through the neighbourhood component analysis[102] (NCA) or adaptations of the base kNN algorithm[52]. However, due to the curse of dimensionality[51] and its effects on metrics such as Euclidean distance[74], it appears that the main algorithms are largely improved through prior dimensionality reduction or metric learning[99].

2 Dimensionality Reduction Methods

Linear dimensionality reduction methods are computationally attractive, and yield intuitive geometric representations of the data[26]; for these reasons alone they often seen as the most fundamental methods for analysing high dimensional data.

Although a number of linear methods are outlined below, a comprehensive review is provided by Cunningham and Ghahramani[26] where the authors expertly compare and contrast these methods and others; as well as identifying their use in preprocessing.

Non-linear dimensionality methods, however, are often more computationally expensive - such as relying on the outputs of optimisation problems. A number of these methods are adaptations of each other, or are inspired from one another. For example, ISOMAP[88] is famously an extension of multidimensional scaling (MDS)[90,37,11], which in turn is the general class of analysis that PCA fits within; in particular the Torgerson MDS on Euclidean distances[90].

2.1 Multidimensional Scaling, and Principal Component Analysis

Multidimensional scaling is often seen as the earliest manifold learning method. The core of MDS implementations is to preserve some metric, often the similarity [90] or dissimilarity[11] between points, in a lower dimensional subspace. Often, methods such as MDS are stated as linear dimensionality reduction methods, but only a subset of cases are. Metric MDS (mMDS)[4], with metrics such as Euclidean distance *are* linear methods. The following algorithm, often poised as an optimisation problem, is the basis of MDS.

The data collection consists of **M** objects, over which a *distance* is defined.
With $d_{i,j}$ = the distance between the i^{th} and j^{th} points.
Then, for each of M objects, a distance matrix D can be constructed:

$$D = \begin{bmatrix} d_{1,1} & d_{1,2} & \dots & d_{1,M} \\ d_{2,1} & d_{2,2} & \dots & d_{2,M} \\ \vdots & \vdots & \ddots & \vdots \\ d_{M,1} & d_{M,2} & \dots & d_{M,M} \end{bmatrix}$$

Thus, given D, find **M** vectors; $\{X_1, X_2, \dots X_M\} \in \mathbb{R}^N$
such that, $\|X_i - X_j\| \approx d_{i,j}$ for all $i, j \in 1, \dots, M$.

It is clear that there are similarities between manifold learning, and MDS. In MDS (Torgerson), the vector norm is the Euclidean distance, however in non-metric MDS this distance can be any 'true' distance or metric[19]. The means to which one finds $\{X_1, X_2, \dots X_M\}$ is often done through an optimisation process[18].

For example, minimising the square loss:

$$\underset{X_1,\dots,X_M}{\operatorname{argmin}} \sum_{i<j} (\|X_i - X_j\| - d_{i,j})^2.$$

Or in the case of Torgerson's method, minimising the 'Strain':

$$\text{Strain}_D(X_1, X_2, ..., X_N) := \left(\frac{\sum_{i,j} \left(a_{ij} - X_i^T X_j \right)^2}{\sum_{i,j} a_{i,j}^2} \right)^{\frac{1}{2}}$$

Where a are the elements of A, which is the double centered matrix of D.

In principal component analysis (PCA), we compute the linear projections of greatest variance from the eigenvectors of the data covariance matrix with greatest corresponding eigenvalues[15]. From this, the principal components (PCs) can be used to construct a lower dimensional embedding, with the weights of the PCs representing the percentage of variance explained[92]. If fewer PCs can explain a desired amount of variance compared to the full dimensional space of the data, PCA can become a suitable dimensionality reduction method.

As previously mentioned, the Euclidean nature of PCA means that topological spaces are not faithfully preserved compared to non-Euclidean methods. The success of PCA as a dimensionality reduction method therefore largely depends on the dataset, its ability to be mapped into Euclidean space, and the relative weighting of the PCs. Methods, examples and commentary on PCA is abundant, and the reader is recommended to the following references[46,47].

2.2 Conventional Methods from Machine Learning

Machine learning methods for manifold learning are more modern and harder to interpret, but are at times able to achieve much better metrics (such as smaller reconstruction errors) than more traditional methods[8,21,22].

Locally linear embedding (LLE)[77] is a method designed to preserve the non-linear structures of datasets (such as manifolds), through serial linear reconstructions[38].

LLE is often regarded as a series of local PCAs designed to preserve global structure in a dataset[77].

Many dimensionality reduction methods have significant time complexity or space complexity[101], whereas LLE provides a computationally efficient method for preserving inherent information encoded within the dataset, in a smaller dimensional space[18].

Although PCA and LLE are both *linear* dimensionality reduction methods, LLE offers preservation of global topology and is therefore a more faithful manifold learning representation[69]. However, due to the linear nature of the reconstruction, LLE fails to preserve complex structure as well as other methods[101]. Although LLE and adapted methods (such as Hessian LLE[95]) sit within the field of machine learning (in addition to manifold learning) there are superior methods involving more modern techniques such as neural networks[100,89,56].

Autoencoders are a key example of manifold learning[13]. In essence, autoencoders are a type of artificial neural network (ANN) that compress the data set say, $\mathbf{M} \in \mathbb{R}^N$, into a latent space of dimension n where typically $n << N$[73]. This is achieved by training both an encoder stage and decoder stage either side of a latent space compression (both stages possibly consisting of multiple layers)[40,21,24,55]. The model is then trained with reconstruction as a loss function (typically as Mean Squared Error (MSE)). A key strength of autoencoders is their ability to embed new data, without refitting a model to the entire dataset[53].

The resultant latent space can be treated as an approximate manifold learning. That is, autoencoders can be used for manifold learning, but not all autoencoders are used for manifold learning; and similarly most manifold learning algorithms are not autoencoders. Autoencoders are comparable to PCA and other methods under certain architectural designs[33]. For example, a single layer network with a linear activation function will find a PCA approximate of up to 'n' components if the latent space is n-dimensional[75]. However, considerable drawbacks include the fact there is no guarantee that components are orthogonal (as is the case with PCA), and the difficulty in interpretation with machine learning outputs, as well as significant computational cost compared to other methods.

Another machine learning method is the 'ivis' method[84]. Utilising a siamese network, with a modified triplet loss[30] - a method that relies on the distances between an anchor point, and a positive and negative point as part of the minimisation function. This triplet loss approach is also utilised in discrete manifold approximations such as TriMap[7].

2.3 Discrete Manifold Approximations

Methods such as Uniform Manifold Approximation and Projection (UMAP)[68], **Iso**metric **Map**ping (ISOMAP)[88], T-distributed Stochastic Neighbour Embeddings (tSNE)[61] and others are graph based algorithms, where nodes are points in higher dimensional space (often observations), and where edges often represent some weighted presence (or absence) of similarity.

ISOMAP was the first of these methods, and was one of the most popular methods for non-linear dimensionality reduction until the inception of tSNE. ISOMAP relies on an extension of MDS based on the geodesic distances as determined by a kNN graph. UMAP also utilises a kNN-like graph (a fuzzy topological representation[68]), but includes an optimisation algorithm to preserve the distances in this graph into a lower-dimensional space (defaulting to Euclidean geometry). The tSNE approach also uses a nearest neighbour graph, but the connectivity is based on conditional probabilities from a Guassian function centred at each data point.

Often, given such a high-dimensional graph, one must find a loss function to accurately preserve structure in a lower dimensional embedding. This commonality was explored in the PaCMAP[97] algorithm. Although many comparisons can be made between these algorithms, such as the equivalence of ISOMAP to a contagion map at $t = 0$[87], the algorithms covered within this paper all have differing approaches.

In addition, comprehensive frameworks such as Minimum Distortion Embedding (MDE), as described by Agrawal, Ali and Boyd, cover embedding tasks, such as those covered in 'ivis' and 'UMAP' and extend upon work by comprehensively evaluating distortion, various loss functions and other important parameters of similar algorithms[5].

3 Dimensionality Reduction for Discovery

Typically, dimensionality reduction is done for a number of reasons:

- to improve the computational costs (less time, less storage) associated with a data set[18];
- to reduce the effect of correlation or collinearity[50,83];
- to remove or reduce the effects of 'noise'[16];
- for visualisation.

All of the above reasons are desirable so that the data can ultimately be used in some type of algorithm or analysed in some way. Commonplace is dimensionality reduction in genetic data, where expression sets can easily include hundreds of thousands of genes, across multiple time points and treatment arms[100,103,9]. At times, methods such as Whole Genome Correlation Network Analysis (WGCNA)[54] are employed for this data for novel gene discovery, or to investigate and discover points of interest in a dataset. The place of dimensionality reduction in knowledge discovery outside of use as a data pre-processing step is beginning to be understood[46], but more research is necessary[63,62,45].

Network based methods appear to provide the most faithful representations of the local and global topology of datasets based on current research. Naturally, methods relying on spectral analysis, including network analysis (WGCNA and others[60,80]) share approaches with network-based methods of manifold learning. Therefore, explicit manifold learning techniques are likely able to be used for discovery in datasets.

3.1 ManiNetCluster

ManiNetCluster is a series of algorithms and techniques demonstrated by Nguyen et al.[71] for use in investigating functional relationships in temporal (longitudinal) gene expression datasets, and serves as one of (if not *the*) first methods designed to investigate data in this way with manifolds.

The method proposed in the paper utilises k-nearest neighbours as a means to construct a graph representation of a manifold, much like UMAP or ISOMAP. By default, this method utilises Euclidean distances to determine if nodes (data points) are connected (edges)[29]. As identified earlier, for most kNN implementations, 'k' is a user-defined hyperparemeter and can have significant effects on the graph and downstream outputs[29].

The algorithm was then, in essence, used to construct a graph for each time point in a dataset. Manifold warping and alignment[93] is then used to match manifolds over time, and the resultant projection then clustered. Clusters are then functionally labelled based on prior groupings (increased over time, decreased over time etc.). The authors of ManiNetCluster and others[97,37,32,35] have developed important ideas in the realm of manifold discovery. Firstly, many manifold learning methods produce differing, but at times comparable results. Also, that using multiple methods can bolster the ability to investigate a dataset. For example, as noted by Nguyen et al., by utilising their method *in addition* to WGCNA we can "[find] that functional linkage modules generated by ManiNetCluster can connect multiple WGCNA modules," - which help explain in a higher-dimensional context the relationship of the dataset. It is evident that neither method in isolation would create these findings as per the author's implementations[9,14,15,31,66]. In their paper the time-matching manifold warping appears to reduce the Guttman effect, that is, the temporal effect in the embedding[96,86] (or at least not in the form they commonly appear as), as shown in Figure 1).

4 A Novel Manifold-Based Method for Temporal Data

4.1 Example Dataset

An example Microexpression array dataset is used throughout this section. The dataset contains seven different experiments (including control). Each experiment has three replicates. Each replicate is based on 150 individually grown plants, blended together. The dataset has approximately 1600 significantly differential genes (log2fold).

4.2 Horseshoe Effects in Temporal Data

Ecological profiles or gene expression datasets are particularly sparse, stymieing traditional statistics. The Guttman effect (often known as the 'horseshoe' or 'arch' effects) often occurs when there is a single gradient, or direction of variance in the data[28]. This can occur when the distance metric causes saturation, as

proposed to be distances such as UniFrac, Euclidean and others commonly used in biological contexts [70]. Latent ordering within the data, or indistinguishable elements in a dataset may also cause this distortion[44].

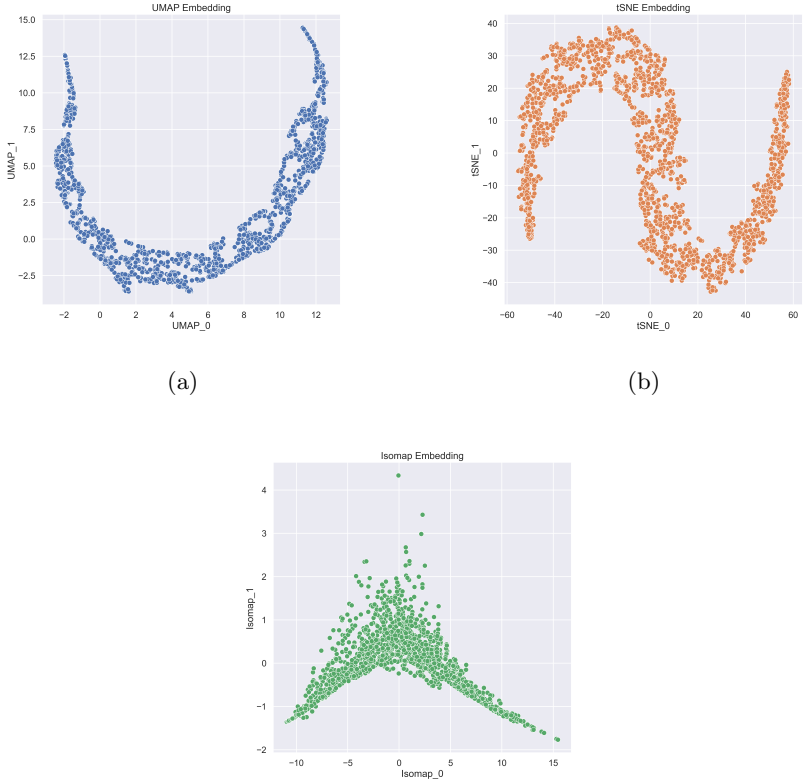
Methods were created to detrend the Guttman effect in data, such as detrending correspondence analysis[41], but there have been criticisms of these methods in the way they may distort the underlying gradients, curves or patterns in the data[34].

By determining an appropriate, non-saturating metric, the horseshoe effect can be eliminated or reduced. The art of identifying and resolving this effect is best described by Morton et al.[70]. Further research is needed to determine the saturation of metrics from metric learning, machine learning algorithms.

Manifold embedding methods, including those mentioned previously, typically expose an array of metrics to utilise in their programmatic implementations. For example, the 'umap-learn' package for Python 3.6[67] provides metrics in various classes such as;

- Minkowski style metrics - e.g. Euclidean and Manhattan;
- spatial metrics - e.g. Canberra, Bray-Curtis;
- binary metrics - e.g. Hamming, Jacard.

However, by default, many manifold embedding methods default to Euclidean distance - a saturating metric - and as a result the Guttman effect can be profound in the latent embeddings as shown in Figure 1. The prevalence and significance of this effect is often overlooked in manifold analysis research and requires more understanding in terms of downstream analysis outputs (i.e. the effect on clustering)[98,94].



(a) (b)

(c)

Fig. 1: Embedding with default parameters in (a) UMAP, (b) tSNE and (c) Isomap on a dataset with strong batch effects results in significant Guttman effect in two-dimensional embeddings.

An ideal manifold discovery method therefore either utilises a non-saturating metric (for a given dataset) or is somehow invariant to the Guttman effect. This desired invariance can be achieved if the model is sufficiently *robust*, however, even modern methods such as TriMAP face difficulty in dimensionality reduction when outliers or noise are present[7]. Nevertheless, the manifold warping methods utilised in ManiNetCluster[93], the AlignedUMAP variant[1] of UMAP[68], or methods such as 2-MAP/ThruMAP[59] appear to provide methods to overcome the temporal effects seen by naively utilising longitudinal datasets.

4.3 Spectral Analysis on Manifolds

In a similar fashion to how the authors of ManiNetCluster were able to combine their time-matched manifold representations with the outputs of WGCNA, it was hypothesised that manifold embeddings could be utilised with spectral analysis.

Spectral analysis on graphs for knowledge discovery is a well-established field[80]. Methods such as the graph Laplacian[80], Spectral Clustering and Fiedler vector partitioning[81] have already been shown to be efficient methods for knowledge discovery in gene expression networks[49,81].

Importantly, the Fiedler vector method of spectral analysis utilises the eigenvector of the second smallest eigenvalue to partition connected graphs[81]. Analyses based on the properties of the Fiedler vector already have been shown to be a mathematically robust method to identify clear partitions in bionformatic data[20,2]. The use of Fiedler vectors in analysis typically requires connected graphs, which methods such as kNN cannot guarantee by default. An alternative is to utilise subgraph partitioning, to isolate and individually analyse subgraphs, or to connect all disconnected graphs to ensure that the graph is connected.

4.4 Proposed Algorithm Overview

The following sections outline a general framework for reducing the Guttman effect in manifolds, for example, from temporal effects in longitudinal gene expression datasets. By doing so, we hope to reduce both local and global distortions and create more faithful manifold representations. In addition, we aim to highlight the ability to utilise spectral analysis methods on network-based dimensionality reduction or manifold learning methods.

Manifold alignment can be done via warping if the assumption of datasets lying on the same manifold can be taken. Formally[43,64]

consider two datasets:

$$X, Y : X_i \in \mathbb{R}^m, Y_i \in R^n.$$

'Manifold Alignment' aims to find projections, ϕ, so that:

$$\phi_X : \mathbb{R}^m \to \mathbb{R}^d$$
$$\phi_Y : \mathbb{R}^n \to \mathbb{R}^d.$$

Then given some known correspondence (binary):

$$W_{ij} = \begin{cases} 1 & X_i \Longleftrightarrow Y_i \\ 0 & else, \end{cases}$$

with within-dataset distances (often kNN graph):S_X, S_Y

A manifold preservation fidelity: μ

Then the loss function is given as:

$$\arg\min_{\phi_X, \phi_Y} \mu \sum_{i,j} \left\| \phi_X\left(X_i\right) - \phi_X\left(X_j\right) \right\|^2 S_{X,i,j} + \ldots$$

$$\mu \textstyle\sum_{i,j} \left\| \phi_Y\left(Y_i\right) - \phi_Y\left(Y_j\right) \right\|^2 S_{Y,i,j} + (1-\mu) \sum_{i,j} \left\| \phi_X\left(X_i\right) - \phi_Y\left(Y_j\right) \right\|^2 W_{i,j}.$$

Famously, Belkin et al.[12] note this is equivalent to solving a generalised eigenvalue problem (where G is the Graph Laplacian);

$$G = \begin{bmatrix} \mu S_X & (1-\mu) W \\ (1-\mu) W^T & \mu S_Y \end{bmatrix}.$$

For temporal data, a proposed alteration by Vu et al.[93] is utilised. Both methods extend to any number of datasets (including time points). The resultant

manifold is a reduced projection that can provide insight into how the same data (such as genes) behave over time or in discrete datasets.

By creating aligned manifold projections (such as unions, AlignedUMAP[1], ManiNetCluster[71], Vu[93] or similar), effects due to time or global gradients appear to be reduced. However, these methods often create a disconnected graph. Therefore, downstream analysis should be aware and invariant to this if it occurs.

The proposed algorithms involve creating local partitions (or subspaces) of the manifold for analysis. Therefore, some method must be used to create a semi-optimal nearest-neighbour embedding for use. Popular implementations include ANNOY[3] and PyNNDescent[29]; although a number of alternative methods exist and further research is necessary to evaluate all methods. Although, many methods are implementation-specific or problem-specific[57], further refinement of this component is outside the scope of this work. In addition, these manifolds are then aligned over time (if the dataset is temporal) to reduce the effect of the Guttman effect, as implemented in ManiNetCluster[71], or as recent method 'Multiscale Manifold Warping'[65].

Methods that do not create submodules but utilise kNN graphs (such as UMAP - which uses PyNNDescent) can be used to create the aligned graphs, which can than be broken down into constituent subgraphs of connected components. Once a graph is built, sub-networks (here treated as distinct subspaces) can be calculated from disconnected components, weakly connected components[85], strongly connected components[72] or by using some other graph-based algorithm for detecting subgraphs.

The metric for the network embedding is by default Euclidean distance, but can be any supported distance by UMAP, including custom metrics as per implementation in 'umap-learn'[68]. The neighbourhood connectivity is by default the minimum spanning tree (MST) 'all' variant as described by Dalmia et al.[27] to improve the global and local preservation. In addition, MST preprocessing improves clustering and downstream outputs[27]. Example algorithmic approaches for various analyses are shown below.

Although methods such as UMAP are capable of creating aligned manifolds, the proposed algorithm ensures downstream spectral analysis is possible, and reduces the Guttman effects.

Algorithm 1 Clustering Temporal Aligned Manifolds

Require: M: a matrix where one dimension is time.

for each time point in M as n **do**
    $G_n \leftarrow construct\_knn\_graph(M)$ ▷ Create kNN graph.[68]
end for
$G_{aligned} \leftarrow align\_manifolds(M_0, \ldots, M_n)$ ▷ Create aligned graph[93]
$embedding \leftarrow umap(G_{aligned})$ ▷ Create embedding from custom kNN graph[67]
$clusters \leftarrow cluster(embedding)$ ▷ Use embedding as preprocessing for downstream pipelines

Algorithm 2 Partitioning Time Aligned Manifolds

1: **for** each time point in M as n **do**
2: $G_n \leftarrow construct\_knn\_graph(M)$
3: **end for**
4: $G_{aligned} \leftarrow align\_manifolds(M_0, \ldots, M_n)$
5: $G_0, G_1 = fiedler\_partition(G_{aligned})$ ▷ Use Graph Laplacian to identify Fiedler vectors[81]

Example implementations of the above algorithms are provided in Python 3.8.10 (Section **??**).

4.5 Example Algorithm Implemention

An example dataset based on expression of Arabidopsis *thaliana* was used to compare the ability of the algorithm to perform biologically intuitive dimensionality reduction, compared to a common algorithm - UMAP. The data is a set of microarray expression experiments, the utilised experiment is a decapitation, where a set of controls is compared to the expression of plants that has had material cut from them. There are a total of 1644 genes with an absolute fold change greater than or equal to two. The data was transformed using a standard scaling, so that each gene expression was centred around 0, with a minimum of -1, and a maximum of 1.

The experimental gene expressions are averaged over 150 technical replicates for both control and experiment (decapitation). An aligned manifold projection is created to unify the manifold of the experimental gene expressions, with those of the controls. This analysis aims to identify which genes behave similarly compared to the control expression.

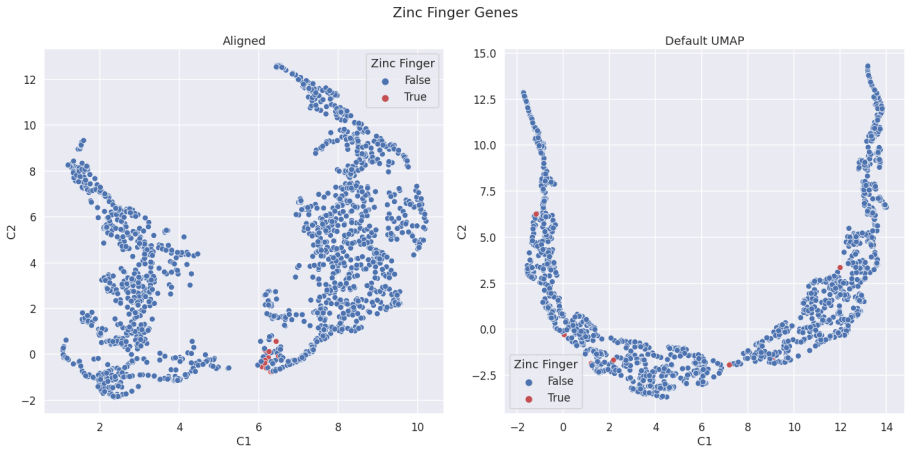


Fig. 2: Zinc Finger genes (N = 15), appear proximal to each other in the altered projection, but appear uniformly distributed in the UMAP projection.

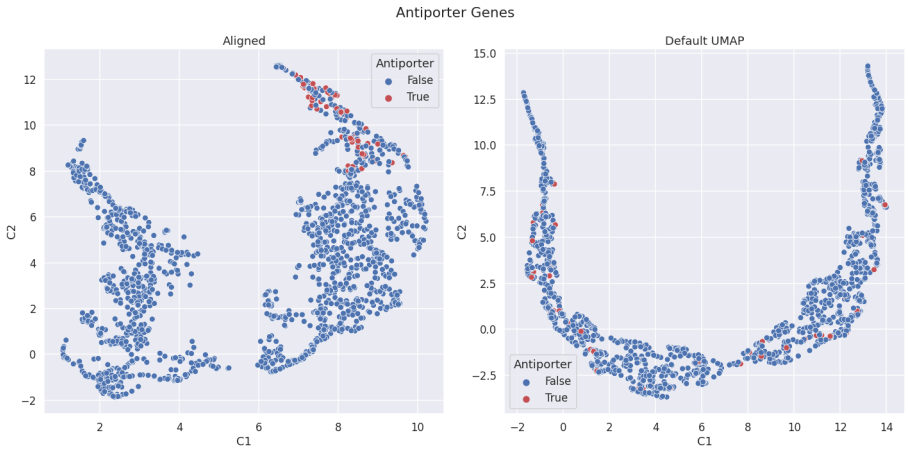


Fig. 3: Antiporter genes (N = 60), appear to be clustered in a single 'half'/'island' of the projection. The genes are distributed towards the lower half of the UMAP projection.

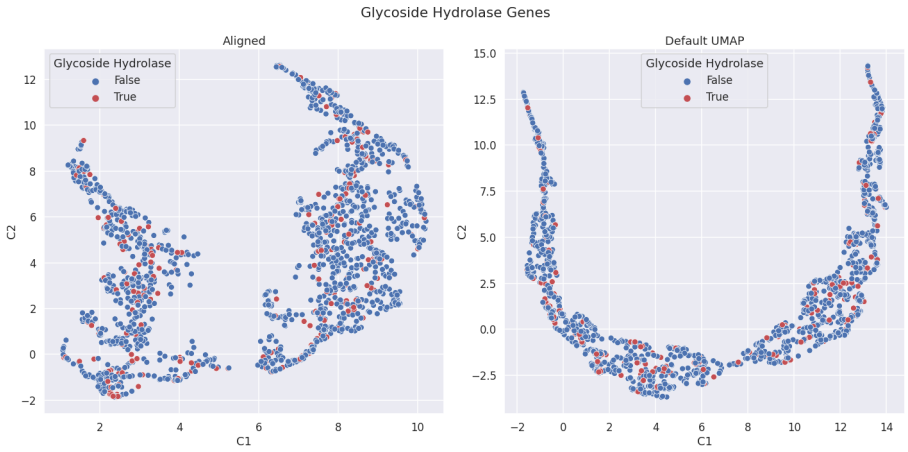


Fig. 4: The Glycoside Hydrolase genes (N = 223) appear not to cluster in either method, possibly implying the genes do not behave/express similarly in a biological context.

The resultant lower-dimensional projection of the algorithm (left subfigures) appear to be effected by the Guttman effect much less than the base UMAP projection (right subfigures), if at all. Genes that are close together behave similarly compared to their behaviour against the control expression. That is to say, for example, the Zinc Finger genes all express similarly to each other, in the way in which they express compared to the control expression.

5 Conclusion

Manifold learning, although a recent field in the realm of Mathematics, has seen significant interest in the last two decades. Although there was a quiet period after Tenenbaum's seminal ISOMAP algorithm, recent interest has piqued thanks to innovative algorithms of manifold learning such as UMAP, tSNE and Ivis among others.

Methods combining manifold learning, such as spectral analysis (such as WGCNA) appear to enable researchers to gain insight into non-linear relationships in highly dimensional datasets. Datasets that would otherwise be prone to issues such as the Guttman effect. In addition, recent advancements such as manifold warping[65] and alignment[93] provide standardised methods to enable the analysis of networks sharing dependent dimensions such as time.

Although a number of manifold dimensionality reduction methods are able to create intuitive visualisations, or usable embeddings for example datasets, there is still much research necessary in reducing or eliminating the effects of noise - as several methods identify noise or outliers to be the most significant effect on the resultant latent space embeddings[7].

The methodology described within this paper provides a general framework to utilise aligned manifolds in a variety of settings, such as in visualisation, clustering or spectral analysis problems.

6 Data Availability

All code and data to recreate the figures in this paper, as well as examples are available at https://github.com/Plant-Success/ManiLWA.

7 Author Contributions

The authors (A.G.H., K.B.) contributed equally to this article.

8 Competing Interests

The authors declare no competing interests. The research was funded by the Australian Research Centre, Centre of Excellence for Plant Success in Nature and Agriculture.

9 Acknowledgements

The authors would like to acknowledge Dr. Alicia Hellens and Professor Christine Beveridge for providing the dataset. We would like to acknowledge Professor Peter Waterhouse and his group, in particular Ms. Zuba Ahmed for their guidance with the analysis.

References

1. AlignedUMAP for Time Varying Data — umap 0.5 documentation, https://umap-learn.readthedocs.io/en/latest/aligned_umap_politics_demo.html
2. Spectral Graph Theory (CBMS Regional Conference Series in Mathematics, No. 92): Fan R. K. Chung: 9780821803158: Books - Amazon, https://www.amazon.com/exec/obidos/ASIN/0821803158/ref=nosim/ericstreasuretro
3. spotify/annoy (Mar 2022), https://github.com/spotify/annoy, original-date: 2013-04-01T20:29:40Z
4. Agarwal, S., Wills, J., Cayton, L., Lanckriet, G., Kriegman, D., Belongie, S.: Generalized Non-metric Multidimensional Scaling p. 8
5. Agrawal, A., Ali, A., Boyd, S.: Minimum-Distortion Embedding. Foundations and Trends® in Machine Learning **14**(3), 211–378 (2021). https://doi.org/10.1561/2200000090, http://www.nowpublishers.com/article/Details/MAL-090
6. Al-Tashi, Q., Abdulkadir, S.J., Rais, H.M., Mirjalili, S., Alhussian, H.: Approaches to Multi-Objective Feature Selection: A Systematic Literature Review. IEEE Access **8**, 125076–125096 (2020). https://doi.org/10.1109/ACCESS.2020.3007291, conference Name: IEEE Access
7. Amid, E., Warmuth, M.K.: TriMap: Large-scale Dimensionality Reduction Using Triplets. arXiv:1910.00204 [cs, stat] (Oct 2019), http://arxiv.org/abs/1910.00204, arXiv: 1910.00204
8. Araya, R.E., Goldszmid, R.S.: Chapter Seventeen - Characterization of the tumor immune infiltrate by multiparametric flow cytometry and unbiased high-dimensional data analysis. In: Galluzzi, L., Rudqvist, N.P. (eds.) Methods in Enzymology, Tumor Immunology and Immunotherapy – Cellular Methods Part B, vol. 632, pp. 309–337. Academic Press (Jan 2020). https://doi.org/10.1016/bs.mie.2019.11.012, https://www.sciencedirect.com/science/article/pii/S0076687919304690
9. Arowolo, M.O., Adebiyi, M.O., Aremu, C., Adebiyi, A.A.: A survey of dimension reduction and classification methods for RNA-Seq data on malaria vector. Journal of Big Data **8**(1), 50 (Mar 2021). https://doi.org/10.1186/s40537-021-00441-x, https://doi.org/10.1186/s40537-021-00441-x
10. Ayesha, S., Hanif, M.K., Talib, R.: Overview and comparative study of dimensionality reduction techniques for high dimensional data. Information Fusion **59**, 44–58 (Jul 2020). https://doi.org/10.1016/j.inffus.2020.01.005, https://www.sciencedirect.com/science/article/pii/S156625351930377X
11. Beals, R., Krantz, D.H., Tversky, A.: Foundations of multidimensional scaling. Psychological Review **75**(2), 127–142 (1968). https://doi.org/10.1037/h0025470, place: US Publisher: American Psychological Association
12. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. Neural Computation **15**, 1373–1396 (2003)
13. Bengio, Y., Courville, A., Vincent, P.: Representation Learning: A Review and New Perspectives. arXiv:1206.5538 [cs] (Apr 2014), http://arxiv.org/abs/1206.5538, arXiv: 1206.5538
14. Bogomolov, S., Forets, M., Frehse, G., Podelski, A., Schilling, C., Viry, F.: Reach Set Approximation through Decomposition with Low-dimensional Sets and High-dimensional Matrices. In: Proceedings of the 21st International Conference on Hybrid Systems: Computation and Control (part of CPS Week). pp. 41–50 (Apr 2018). https://doi.org/10.1145/3178126.3178128, http://arxiv.org/abs/1801.09526, arXiv:1801.09526 [cs, math]

15. Bouwmans, T., Zahzah, E.H.: Robust PCA via Principal Component Pursuit: A review for a comparative evaluation in video surveillance. Computer Vision and Image Understanding **122**, 22–34 (May 2014). https://doi.org/10.1016/j.cviu.2013.11.009, https://www.sciencedirect.com/science/article/pii/S1077314213002294
16. Braun, M.L., Buhmann, J.: Denoising and Dimension Reduction in Feature Space p. 8
17. Buja, A., Swayne, D.F., Littman, M.L., Dean, N., Hofmann, H., Chen, L.: Data Visualization With Multidimensional Scaling. Journal of Computational and Graphical Statistics **17**(2), 444–472 (Jun 2008). https://doi.org/10.1198/106186008X318440, http://www.tandfonline.com/doi/abs/10.1198/106186008X318440
18. Cadima, J., Cerdeira, J.O., Minhoto, M.: Computational aspects of algorithms for variable selection in the context of principal components. Computational Statistics & Data Analysis **47**(2), 225–236 (Sep 2004). https://doi.org/10.1016/j.csda.2003.11.001, https://www.sciencedirect.com/science/article/pii/S016794730300272X
19. Candès, E.J., Li, X., Ma, Y., Wright, J.: Robust principal component analysis? Journal of the ACM **58**(3), 11:1–11:37 (Jun 2011). https://doi.org/10.1145/1970392.1970395, https://doi.org/10.1145/1970392.1970395
20. Chen, J., Hero, III, A.O., Rajapakse, I.: Spectral identification of topological domains. Bioinformatics **32**(14), 2151–2158 (Jul 2016). https://doi.org/10.1093/bioinformatics/btw221, https://doi.org/10.1093/bioinformatics/btw221
21. Chen, N., Klushyn, A., Ferroni, F., Bayer, J., van der Smagt, P.: Learning Flat Latent Manifolds with VAEs. Tech. Rep. arXiv:2002.04881, arXiv (Aug 2020), http://arxiv.org/abs/2002.04881, arXiv:2002.04881 [cs, stat] type: article
22. Chigirev, D., Bialek, W.: Optimal Manifold Representation of Data: An Information Theoretic Approach p. 8
23. Chu, C.: The Geometry of Data: Distance on Data Manifolds p. 117
24. Connor, M., Canal, G., Rozell, C.: Variational Autoencoder with Learned Latent Structure. In: Proceedings of The 24th International Conference on Artificial Intelligence and Statistics. pp. 2359–2367. PMLR (Mar 2021), https://proceedings.mlr.press/v130/connor21a.html, iSSN: 2640-3498
25. Costa, J., Hero, A.: Manifold learning using Euclidean k-nearest neighbor graphs [image processing examples]. In: 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing. vol. 3, pp. iii–988 (May 2004). https://doi.org/10.1109/ICASSP.2004.1326713, iSSN: 1520-6149
26. Cunningham, J.P., Ghahramani, Z.: Linear Dimensionality Reduction: Survey, Insights, and Generalizations p. 42
27. Dalmia, A., Sia, S.: Clustering with UMAP: Why and How Connectivity Matters. arXiv:2108.05525 [cs] (Dec 2021), http://arxiv.org/abs/2108.05525, arXiv:2108.05525
28. Diaconis, P., Goel, S., Holmes, S.: Horseshoes in multidimensional scaling and local kernel methods. The Annals of Applied Statistics **2**(3), 777–807 (Sep 2008). https://doi.org/10.1214/08-AOAS165, https://projecteuclid.org/journals/annals-of-applied-statistics/volume-2/issue-3/Horseshoes-in-multidimensional-scaling-and-local-kernel-methods/10.1214/08-AOAS165.full, publisher: Institute of Mathematical Statistics

29. Dong, W., Moses, C., Li, K.: Efficient k-nearest neighbor graph construction for generic similarity measures. In: Proceedings of the 20th international conference on World wide web - WWW '11. p. 577. ACM Press, Hyderabad, India (2011). https://doi.org/10.1145/1963405.1963487, http://portal.acm.org/citation.cfm?doid=1963405.1963487
30. Dong, X., Shen, J.: Triplet Loss in Siamese Network for Object Tracking. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) Computer Vision – ECCV 2018, vol. 11217, pp. 472–488. Springer International Publishing, Cham (2018). https://doi.org/10.1007/978-3-030-01261-$8_2$8, http://link.springer.com/10.1007/978-3-030-01261-8_28, series Title: Lecture Notes in Computer Science
31. Dong, Y., Qin, S.J., Boyd, S.P.: Extracting a low-dimensional predictable time series. Optimization and Engineering **23**(2), 1189–1214 (Jun 2022). https://doi.org/10.1007/s11081-021-09643-x, https://link.springer.com/10.1007/s11081-021-09643-x
32. Donoho, D.L.: High-dimensional data analysis: The curses and blessings of dimensionality. In: Ams Conference on Math Challenges of the 21st Century (2000)
33. Duque, A.F., Morin, S., Wolf, G., Moon, K.R.: Extendable and invertible manifold learning with geometry regularized autoencoders. In: 2020 IEEE International Conference on Big Data (Big Data). pp. 5027–5036 (Dec 2020). https://doi.org/10.1109/BigData50022.2020.9378049, http://arxiv.org/abs/2007.07142, arXiv:2007.07142 [cs, stat]
34. Ejrnæs, R.: Can we trust gradients extracted by Detrended Correspondence Analysis? Journal of Vegetation Science **11**(4), 565–572 (2000). https://doi.org/10.2307/3246586, https://onlinelibrary.wiley.com/doi/abs/10.2307/3246586, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.2307/3246586
35. Fefferman, C., Mitter, S., Narayanan, H.: Testing the manifold hypothesis. Journal of the American Mathematical Society **29**(4), 983–1049 (Feb 2016). https://doi.org/10.1090/jams/852, https://www.ams.org/jams/2016-29-04/S0894-0347-2016-00852-4/
36. Gashler, M., Ventura, D., Martinez, T.: Iterative Non-linear Dimensionality Reduction with Manifold Sculpting. vol. 20 (Jan 2007)
37. Ghojogh, B., Ghodsi, A., Karray, F., Crowley, M.: Multidimensional Scaling, Sammon Mapping, and Isomap: Tutorial and Survey. arXiv:2009.08136 [cs, stat] (Sep 2020), http://arxiv.org/abs/2009.08136, arXiv: 2009.08136
38. Ghojogh, B., Ghodsi, A., Karray, F., Crowley, M.: Generative locally linear embedding: A module for manifold unfolding and visualization. Software Impacts **9**, 100105 (Aug 2021). https://doi.org/10.1016/j.simpa.2021.100105, https://www.sciencedirect.com/science/article/pii/S2665963821000403
39. Hajderanj, L., Chen, D., Weheliye, I.: The Impact of Supervised Manifold Learning on Structure Preserving and Classification Error: A Theoretical Study. IEEE Access **9**, 43909–43922 (2021). https://doi.org/10.1109/ACCESS.2021.3066259, conference Name: IEEE Access
40. Ham, J., Lee, D.D., Mika, S., Schölkopf, B.: A kernel view of the dimensionality reduction of manifolds. In: Twenty-first international conference on Machine learning - ICML '04. p. 47. ACM Press, Banff, Alberta, Canada (2004). https://doi.org/10.1145/1015330.1015417, http://portal.acm.org/citation.cfm?doid=1015330.1015417

41. Hill, M.O., Gauch, H.G.: Detrended Correspondence Analysis: An Improved Ordination Technique. In: van der Maarel, E. (ed.) Classification and Ordination: Symposium on advances in vegetation science, Nijmegen, The Netherlands, May 1979, pp. 47–58. Springer Netherlands, Dordrecht (1980). https://doi.org/10.1007/978-94-009-9197-2$_7$, https://doi.org/10.1007/978-94-009-9197-2_7
42. Hinneburg, A., Keim, D.A.: Optimal Grid-Clustering : Towards Breaking the Curse of Dimensionality in High-Dimensional Clustering p. 11
43. Hou, C., Nie, F., Wang, H., Yi, D., Zhang, C.: Learning high-dimensional correspondence via manifold learning and local approximation. Neural Computing and Applications **24**(7-8), 1555–1568 (2013). https://doi.org/10.1007/s00521-013-1369-z
44. Hsu, L.L., Culhane, A.C.: Impact of Data Preprocessing on Integrative Matrix Factorization of Single Cell Data. Frontiers in Oncology **10** (2020), https://www.frontiersin.org/article/10.3389/fonc.2020.00973
45. Huang, S.H., Wulsin, L.R., Li, H., Guo, J.: Dimensionality reduction for knowledge discovery in medical claims database: Application to antidepressant medication utilization study. Computer Methods and Programs in Biomedicine **93**(2), 115–123 (Feb 2009). https://doi.org/10.1016/j.cmpb.2008.08.002, https://linkinghub.elsevier.com/retrieve/pii/S0169260708002058
46. Huang, X., Wu, L., Ye, Y.: A Review on Dimensionality Reduction Techniques. International Journal of Pattern Recognition and Artificial Intelligence **33**(10), 1950017 (Sep 2019). https://doi.org/10.1142/S0218001419500174, https://www.worldscientific.com/doi/abs/10.1142/S0218001419500174, publisher: World Scientific Publishing Co.
47. Jolliffe, I.T., Cadima, J.: Principal component analysis: a review and recent developments. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences **374**(2065), 20150202 (Apr 2016). https://doi.org/10.1098/rsta.2015.0202, https://royalsocietypublishing.org/doi/10.1098/rsta.2015.0202, publisher: Royal Society
48. Kettenring, J.R.: Coping with high dimensionality in massive datasets. WIREs Computational Statistics **3**(2), 95–103 (2011). https://doi.org/10.1002/wics.141, https://onlinelibrary.wiley.com/doi/abs/10.1002/wics.141, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/wics.141
49. Kim, C., Cheon, M., Kang, M., Chang, I.: A simple and exact Laplacian clustering of complex networking phenomena: Application to gene expression profiles. Proceedings of the National Academy of Sciences **105**(11), 4083–4087 (Mar 2008). https://doi.org/10.1073/pnas.0708598105, https://www.pnas.org/doi/10.1073/pnas.0708598105, publisher: Proceedings of the National Academy of Sciences
50. Kim, H.J., Adluru, N., Bendlin, B.B., Johnson, S.C., Vemuri, B.C., Singh, V.: Canonical Correlation Analysis on Riemannian Manifolds and Its Applications. Computer vision - ECCV ... : ... European Conference on Computer Vision : proceedings. European Conference on Computer Vision **8690**, 251–267 (2014). https://doi.org/10.1007/978-3-319-10605-2$_1$7, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4194269/
51. Kouiroukidis, N., Evangelidis, G.: The Effects of Dimensionality Curse in High Dimensional kNN Search. In: 2011 15th Panhellenic Conference on Informatics. pp. 41–45 (Sep 2011). https://doi.org/10.1109/PCI.2011.45

52. Kramer, O.: Dimensionality Reduction by Unsupervised K-Nearest Neighbor Regression. In: 2011 10th International Conference on Machine Learning and Applications and Workshops. vol. 1, pp. 275–278 (Dec 2011). https://doi.org/10.1109/ICMLA.2011.55
53. Ladjal, S., Newson, A., Pham, C.H.: A PCA-like Autoencoder. Tech. Rep. arXiv:1904.01277, arXiv (Apr 2019). https://doi.org/10.48550/arXiv.1904.01277, http://arxiv.org/abs/1904.01277, arXiv:1904.01277 [cs] type: article
54. Langfelder, P., Horvath, S.: WGCNA: an R package for weighted correlation network analysis. BMC Bioinformatics **9**(1), 559 (Dec 2008). https://doi.org/10.1186/1471-2105-9-559, https://doi.org/10.1186/1471-2105-9-559
55. Lannan, N., Fan, G.: Filter Guided Manifold Optimization in the Autoencoder Latent Space. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). pp. 949–955. IEEE, Long Beach, CA, USA (Jun 2019). https://doi.org/10.1109/CVPRW.2019.00125, https://ieeexplore.ieee.org/document/9025685/
56. Lei, N., Luo, Z., Yau, S.T., Gu, D.X.: Geometric Understanding of Deep Learning. Tech. Rep. arXiv:1805.10451, arXiv (May 2018), http://arxiv.org/abs/1805.10451, arXiv:1805.10451 [cs, stat] type: article
57. Li, W., Zhang, Y., Sun, Y., Wang, W., Zhang, W., Lin, X.: Approximate Nearest Neighbor Search on High Dimensional Data — Experiments, Analyses, and Improvement (v1.0). arXiv:1610.02455 [cs] (Oct 2016), http://arxiv.org/abs/1610.02455, arXiv: 1610.02455
58. Liu, H., Motoda, H.: Feature Extraction, Construction and Selection: A Data Mining Perspective. Springer Science & Business Media (Aug 1998), google-Books-ID: zi_0EdWW5fYC
59. Liu, X., Zhang, Z., Xuan, H., Leontie, R., Stylianou, A., Pless, R.: 2-MAP: Aligned Visualizations for Comparison of High-Dimensional Point Sets p. 9
60. Lunga, D., Prasad, I.S., Crawford, M., Ersoy, O.: Manifold Learning Based Feature Extraction For Classification Of Hyperspectral Data p. 11
61. van der Maaten, L., Hinton, G.: Viualizing data using t-SNE. Journal of Machine Learning Research **9**, 2579–2605 (Nov 2008)
62. Magdalinos, P., Doulkeridis, C., Vazirgiannis, M.: K-Landmarks: Distributed Dimensionality Reduction for Clustering Quality Maintenance. In: Fürnkranz, J., Scheffer, T., Spiliopoulou, M. (eds.) Knowledge Discovery in Databases: PKDD 2006. pp. 322–334. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg (2006). https://doi.org/10.1007/11871637$_3$2
63. Magdalinos, P., Vazirgiannis, M., Valsamou, D.: Distributed Knowledge Discovery with Non Linear Dimensionality Reduction. In: Zaki, M.J., Yu, J.X., Ravindran, B., Pudi, V. (eds.) Advances in Knowledge Discovery and Data Mining. pp. 14–26. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13672-6$_2$
64. Mahadevan, S., Rao, A., Theocharous, G., Healey, J.: Multiscale manifold warping. CoRR **abs/2109.09222** (2021), https://arxiv.org/abs/2109.09222
65. Mahadevan, S., Rao, A., Theocharous, G., Healey, J.: Multiscale Manifold Warping. arXiv:2109.09222 [cs] (Sep 2021), http://arxiv.org/abs/2109.09222, arXiv: 2109.09222
66. Mazher, A.: Visualization Framework for High-Dimensional Spatio-Temporal Hydrological Gridded Datasets Using Machine-Learning Techniques. Water Law **12**, 590 (Feb 2020). https://doi.org/10.3390/w12020590

67. McInnes, L.: lmcinnes/umap (Mar 2022), https://github.com/lmcinnes/umap, original-date: 2017-07-02T01:11:17Z
68. McInnes, L., Healy, J., Melville, J.: UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. arXiv:1802.03426 [cs, stat] (Sep 2020), http://arxiv.org/abs/1802.03426, arXiv: 1802.03426
69. Moor, M., Horn, M., Rieck, B., Borgwardt, K.: Topological Autoencoders. Tech. Rep. arXiv:1906.00722, arXiv (May 2021), http://arxiv.org/abs/1906.00722, arXiv:1906.00722 [cs, math, stat] type: article
70. Morton, J.T., Toran, L., Edlund, A., Metcalf, J.L., Lauber, C., Knight, R.: Uncovering the Horseshoe Effect in Microbial Analyses. mSystems **2**(1), e00166–16 (Feb 2017). https://doi.org/10.1128/mSystems.00166-16, https://journals.asm.org/doi/10.1128/mSystems.00166-16
71. Nguyen, N.D., Blaby, I.K., Wang, D.: ManiNetCluster: a novel manifold learning approach to reveal the functional links between gene networks. BMC Genomics **20**(S12), 1003 (Dec 2019). https://doi.org/10.1186/s12864-019-6329-2, https://bmcgenomics.biomedcentral.com/articles/10.1186/s12864-019-6329-2
72. Nuutila, E., Soisalon-soininen, E.: On Finding the Strongly Connected Components in a Directed Graph. Information Processing Letters **49** (1994)
73. Park, E.: Manifold Learning with Variational Auto-encoder for Medical Image Analysis p. 5
74. Pestov, V.: Is the k-NN classifier in high dimensions affected by the curse of dimensionality? Tech. Rep. arXiv:1110.4347, arXiv (Sep 2012). https://doi.org/10.48550/arXiv.1110.4347, http://arxiv.org/abs/1110.4347, arXiv:1110.4347 [stat] type: article
75. Plaut, E.: From Principal Subspaces to Principal Components with Linear Autoencoders. Tech. Rep. arXiv:1804.10253, arXiv (Dec 2018), http://arxiv.org/abs/1804.10253, arXiv:1804.10253 [cs, stat] type: article
76. Puchkin, N., Spokoiny, V., Stepanov, E., Trevisan, D.: Reconstruction of manifold embeddings into Euclidean spaces via intrinsic distances. Tech. Rep. arXiv:2012.13770, arXiv (Dec 2020). https://doi.org/10.48550/arXiv.2012.13770, http://arxiv.org/abs/2012.13770, arXiv:2012.13770 [math] type: article
77. Roweis, S.T., Lawrence, L.K.: Nonlinear Dimensionality Reduction by Locally Linear Embedding. Science **290**(5500) (2000), https://pub.uni-bielefeld.de/record/2714741
78. Schuster, V., Krogh, A.: A Manifold Learning Perspective on Representation Learning: Learning Decoder and Representations without an Encoder. Entropy **23**(11), 1403 (Oct 2021). https://doi.org/10.3390/e23111403, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8625121/
79. Shahbazi, M., Shirali, A., Aghajan, H., Nili, H.: Using distance on the Riemannian manifold to compare representations in brain and in models. NeuroImage **239**, 118271 (Oct 2021). https://doi.org/10.1016/j.neuroimage.2021.118271, https://www.sciencedirect.com/science/article/pii/S1053811921005474
80. Shuman, D.I., Narang, S.K., Frossard, P., Ortega, A., Vandergheynst, P.: The Emerging Field of Signal Processing on Graphs: Extending High-Dimensional Data Analysis to Networks and Other Irregular Domains. IEEE Signal Processing Magazine **30**(3), 83–98 (May 2013). https://doi.org/10.1109/MSP.2012.2235192, http://arxiv.org/abs/1211.0053, arXiv: 1211.0053
81. Slininger, B.: Fiedler's Theory of Spectral Graph Partitioning
82. Song, F., Guo, Z., Mei, D.: Feature Selection Using Principal Component Analysis. In: Engineering Design and Manufacturing Informatization 2010 In-

ternational Conference on System Science. vol. 1, pp. 27–30 (Nov 2010). https://doi.org/10.1109/ICSEM.2010.14
83. Suri, S., Bailis, P.: DROP: Dimensionality Reduction Optimization for Time Series. In: Proceedings of the 3rd International Workshop on Data Management for End-to-End Machine Learning - DEEM'19. pp. 1–10 (2019). https://doi.org/10.1145/3329486.3329490, http://arxiv.org/abs/1708.00183, arXiv:1708.00183 [cs]
84. Szubert, B., Cole, J.E., Monaco, C., Drozdov, I.: Structure-preserving visualisation of high dimensional single-cell datasets. Scientific Reports **9**(1), 8914 (Jun 2019). https://doi.org/10.1038/s41598-019-45301-0, https://www.nature.com/articles/s41598-019-45301-0, number: 1 Publisher: Nature Publishing Group
85. Tarjan, R.: Depth-First Search and Linear Graph Algorithms. SIAM Journal on Computing **1**(2), 146–160 (Jun 1972). https://doi.org/10.1137/0201010, https://epubs.siam.org/doi/10.1137/0201010, publisher: Society for Industrial and Applied Mathematics
86. Taveira De Souza, J., Carlos De Francisco, A., Carla De Macedo, D.: Dimensionality Reduction in Gene Expression Data Sets. IEEE Access **7**, 61136–61144 (2019). https://doi.org/10.1109/ACCESS.2019.2915519, conference Name: IEEE Access
87. Taylor, D., Klimm, F., Harrington, H.A., Kramár, M., Mischaikow, K., Porter, M.A., Mucha, P.J.: Topological data analysis of contagion maps for examining spreading processes on networks. Nature Communications **6**(1), 7723 (Nov 2015). https://doi.org/10.1038/ncomms8723, http://www.nature.com/articles/ncomms8723
88. Tenenbaum, J.B., Silva, V.d., Langford, J.C.: A Global Geometric Framework for Nonlinear Dimensionality Reduction. Science **290**(5500), 2319–2323 (Dec 2000). https://doi.org/10.1126/science.290.5500.2319, https://www.science.org/doi/10.1126/science.290.5500.2319
89. Tomar, V.S., Rose, R.C.: Graph based manifold regularized deep neural networks for automatic speech recognition. Tech. Rep. arXiv:1606.05925, arXiv (Jun 2016). https://doi.org/10.48550/arXiv.1606.05925, http://arxiv.org/abs/1606.05925, arXiv:1606.05925 [cs, stat] type: article
90. Torgerson, W.S.: Multidimensional scaling of similarity. Psychometrika **30**(4), 379–393 (Dec 1965). https://doi.org/10.1007/BF02289530, https://doi.org/10.1007/BF02289530
91. Uddin, S., Haque, I., Lu, H., Moni, M.A., Gide, E.: Comparative performance analysis of K-nearest neighbour (KNN) algorithm and its different variants for disease prediction. Scientific Reports **12**(1), 6256 (Apr 2022). https://doi.org/10.1038/s41598-022-10358-x, https://www.nature.com/articles/s41598-022-10358-x, number: 1 Publisher: Nature Publishing Group
92. Vichi, M., Saporta, G.: Clustering and disjoint principal component analysis. Computational Statistics & Data Analysis **53**(8), 3194–3208 (Jun 2009). https://doi.org/10.1016/j.csda.2008.05.028, https://www.sciencedirect.com/science/article/pii/S0167947308002934
93. Vu, H.T., Carey, C.J., Mahadevan, S.: Manifold Warping: Manifold Alignment Over Time p. 7
94. Wang, H., van der Laan, M.J.: Dimension reduction with gene expression data using targeted variable importance measurement. BMC Bioinformatics **12**(1), 312 (Jul 2011). https://doi.org/10.1186/1471-2105-12-312, https://doi.org/10.1186/1471-2105-12-312

95. Wang, J.: Hessian Locally Linear Embedding. In: Wang, J. (ed.) Geometric Structure of High-Dimensional Data and Dimensionality Reduction, pp. 249–265. Springer, Berlin, Heidelberg (2012). https://doi.org/10.1007/978-3-642-27497-$8_1$3, https://doi.org/10.1007/978-3-642-27497-8_13
96. Wang, Q., Megalooikonomou, V.: A Dimensionality Reduction Technique for Efficient Time Series Similarity Analysis. Information systems **33**(1), 115–132 (Mar 2008). https://doi.org/10.1016/j.is.2007.07.002, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2390719/
97. Wang, Y., Huang, H., Rudin, C., Shaposhnik, Y.: Understanding How Dimension Reduction Tools Work: An Empirical Approach to Deciphering t-SNE, UMAP, TriMAP, and PaCMAP for Data Visualization. arXiv:2012.04456 [cs, stat] (Aug 2021), http://arxiv.org/abs/2012.04456, arXiv: 2012.04456
98. Wang, Z., Qu, L., Xin, J., Yang, H., Gao, X.: A unified distributed ELM framework with supervised, semi-supervised and unsupervised big data learning. Memetic Computing **11** (Sep 2019). https://doi.org/10.1007/s12293-018-0271-8
99. Weinberger, K.: Lecture 2: k-nearest neighbors / Curse of Dimensionality, https://www.cs.cornell.edu/courses/cs4780/2018fa/lectures/lecturenote02_kNN.html
100. Xiang, R., Wang, W., Yang, L., Wang, S., Xu, C., Chen, X.: A Comparison for Dimensionality Reduction Methods of Single-Cell RNA-seq Data. Frontiers in Genetics **12** (2021), https://www.frontiersin.org/article/10.3389/fgene.2021.646936
101. Xu, J., Mu, H., Wang, Y., Huang, F.: Feature genes selection using supervised locally linear embedding and correlation coefficient for microarray classification. Computational and Mathematical Methods in Medicine **2018**, 1–11 (2018). https://doi.org/10.1155/2018/5490513
102. Yang, W., Wang, K., Zuo, W.: Neighborhood Component Feature Selection for High-Dimensional Data. JCP **7**, 161–168 (Jan 2012). https://doi.org/10.4304/jcp.7.1.161-168
103. Zhang, Z., Chen, L., Xu, P., Xing, L., Hong, Y., Chen, P.: Gene correlation network analysis to identify regulatory factors in sepsis. Journal of Translational Medicine **18**(1), 381 (Oct 2020). https://doi.org/10.1186/s12967-020-02561-z, https://doi.org/10.1186/s12967-020-02561-z