Literature Review

dimensionality Reduction: A comparative review (maaten et al, 2009)

Dalyn McCauley

## Introduction

Dimensionality Reduction: A Comparative Review by Maaten et al analyzes several techniques for dimensionality reduction and compares them on the basis of their performance on artificial datasets, natural datasets and computational costs. This is an important piece of literature because many real world datasets are highly dimensional, and often non-linear, but yet the most popular form of dimensionality reduction is still traditional linear tetchiness such as principal component analysis and classical scaling. These methods are not always the best option depending on the geometry and intrinsic properties of the dataset. Maaten et al dissect different reduction techniques and give an exhaustive review of their abilities and shortcomings that will prove useful for many readers when choosing an effective method. The objective of this paper are (1) to investigate to what extent nonlinear dimensionality reduction outperforms the traditional PCA method, and (2) to identify the inherent weaknesses of the twelve nonlinear dimensionality reduction techniques [1]. This literature review will only define five of the thirteen techniques, one from each subsection of the taxonomy presented in the next section. This paper did not contain any visualizations, so many of the images in this have been sourced from other literature.

## Dimensionality Reduction

Dimensionality reduction takes a matrix **X** of n observation and D dimensions and transforms it into a matrix **Y** with d dimensionality, where d << D. The reduced dimensionality should represent the minimum number of parameters needed to account for all the observed properties, and there are many ways of achieving this. Dimensionality reduction is traditionally done via linear techniques such as Principal Component Analysis (PCA), Factor Analysis (FA) and classical scaling. In practice, real world data is often highly dimensional but also commonly nonlinear which may require an alternative reduction technique. Examples of high dimensional artificial datasets can be found in Figure 8 [1]. The techniques discussed in this paper include, Kernel PCA, Isomap, Maximum Variance Unfolding (MVU), Diffusion Maps, Locally linear embedding (LLE), Laplacian Eigenmaps, Hessian LLE, Local Tangent Space analysis (LTSA), Sammon Mapping, Multilayer autoencoders, Locally linear coordination (LLC) and Manifold charting. The dimensionality techniques are classified in the following taxonomy, Figure 1[1]. The techniques are first divided into convex or nonconvex subsections. Convex methods optimize an objective function that does not contain a local optima, where nonconvex methods optimize functions that do contain local optima.

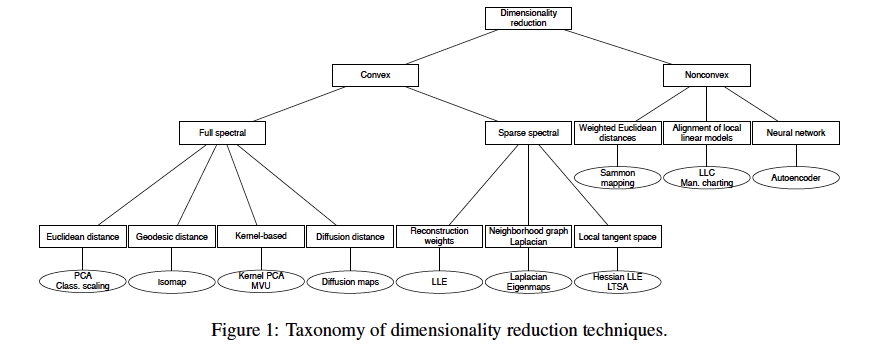


Figure 1

## Convex Techniques for dimensionality reduction

Convex methods are further separated into sparse spectral and full spectral, representing data contained in a sparse matrix and full matrix, respectively. The convex techniques can further be divided into two categories, full spectral and sparse spectral. Full spectral dimensionality reductions use eigen-decompositions of a full matrix that captures the covariance between dimensions or the pairwise similarities between data points and sparse spectral methods use eigen-decomposition of sparse matrices [1].

### PCA/Classical Scaling

Principal component analysis is the most widely used from of dimensionality reduction. PCA performs dimensionality reduction by embedding the data into a linear subspace of lower dimensionality. This done by mapping the high dimensional matrix **X** onto a linear basis **M** by solving for the d principal component eigenvectors of the covariance matrix, Cov(**X**), that describe the directions of highest variance[1]. Mathematically, this is done by maximizing the cost function:

The trace of a matrix is defined as the sum of the main diagonal. Maaten et al suggest PCA has two major drawbacks. The first is that when a data set has high dimensionality, it might be unfeasible to compute all the eigenvectors. In a case where n<<D, classical scaling can be used instead of PCA to overcome this weakness. Secondly, PCA focuses on retaining large pairwise distances instead of focusing on distribution of neighboring data points. For manifolds of high dimensionality and curvature, local geometry and small distances hold important properties that should be maintained in the lower dimensional representation [1].

### Isomap

Isomap methods of dimensionality reduction aim to resolve the problem that arises when datasets fall on a curvilinear manifold, and the distance between two point is much larger than it appears to classical scaling. The swiss roll is a good example of such dataset, Figure 2.

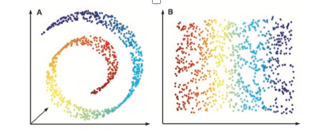


Figure 2

Isomaps preserve geodesic distances between data points. Geodesic distances are defined as the shortest distance between two points on a curved surface. This is done by creating a neighborhood graph G where every data point is connected to its nearest neighbors. The distance between these points are computed and form a pairwise geodesic distance matrix which can then be applied to the highly dimensional **X** matrix using classical scaling [1]. There are three main drawbacks of Isomaps. The first is that they are prone to short circuiting, which can create erroneous data in the neighborhood G matrix. Short circuiting happens when the data contains noise or when the k-nearest neighbors parameter is too large. This can lead to accidental inclusion of points that are close in Euclidean distances but very far away in geodesic distances [2]. This can be visualized in Figure 3 [3], where a large k parameter around the red points might accidentally pick up a green or blue point across the spiral and cause an error in the distance matrix. Secondly, Isomaps are prone to creating holes in the manifold, as seen in image B, Figure 2. Thirdly, Isomaps will not work on nonconvex datasets.



Figure 3

### Kernel PCA

Kernel PCA is similar to traditional PCA but differs in that it first builds a higher dimensional space that is defined by a kernel function. Instead of computing principal components from the covariance matrix such as in traditionally PCA, the kernel PCA computes principle components by solving for the eigenvectors of the kernel matrix, **K** [1]. The most common kernel functions are the linear kernel, polynomial kernel and Gaussian kernel, see Figure 4 [1,4].



Figure 4

The discovery of kernel functions have led to much development in machine learning, specifically in Support Vector Machines and Kernel Ridge Regression. The main disadvantages of kernel PCA is that the kernel matrix is proportional to the square of indices in the dataset and that it mainly focuses on retaining large pairwise distances in the feature space [1]. Additionally, it is often difficult to intelligently select a kernel function without knowing the dimensionality of the original dataset.

### LLE

Local linear embedding is a technique within the non-convex subsection. LLE aims to preserve the topology of the data. It does this by constructing a neighborhood graph, **G**, that preserves solely local properties and allows for successful embedding of non-convex manifolds. The data points are embedded into the lower dimensional space by writing the high dimensional points as a linear combination of their nearest neighbors, known as reconstruction weights [1]. These weights are combined to form a sparse matrix, **W**. Solving for the eigenvectors of the **W** matrix will lead to the low dimensional representation of the data. This process is depicted in Figure 5 [5].



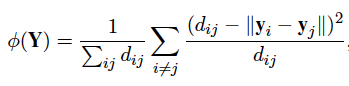
Figure 5

Local linear embedding suffers when used on manifolds with holes and tends to collapse data in the lower dimensional representation. Successful application of LLE include sound source localization and super resolution.

## Non-convex Techniques for dimensionality reduction

### Sammon Mapping

The last dimensionality reduction method I will discuss is Sammon mapping, which is a manifold learning technique. Sammon mapping is one of the non-convex techniques because it’s iterative solving mechanism allows it to learn non-convex datasets. Sammon mapping is a multidimensional scaling technique that adapts the classical scaling by weighting the contribution of each pairwise distance. In doing so, it no longer biases small pairwise distances, and gives a roughly equal weight to retaining all pairwise distances of the high dimensional data. Because of this, Sammon mapping can retain the local structure of the data better than classical scaling. Sammon mapping minimizes the cost function below by iteratively solving for the low dimensional configuration such that the difference in pairwise distances in **X** and **Y** are minimized.



Sammon mapping is mainly used for visualization of data because of how well it can retain the overall structure of the data. Figure 6 shows a good example of how the swill roll is reconstructed to a two dimensional space using Sammon mapping [6]. Notice the difference between this method and the results of Isomaps, Figure 2.



Figure 6

## Results

The authors compared the thirteen dimensionality reduction techniques on the basis of their computational costs and performance on artificial and natural datasets. The metrics of performance used are the generalization error, the trustworthiness and the continuity. Maaten et al define trustworthiness as the evaluation of the set of points that are among the k-nearest neighbors in the low dimensional space but not in the high dimensional space. Conversely, the continuity is the evaluation of the set of points that are among the k-nearest neighbors in the high dimensional space but not in the low dimensional space.

### Computational Costs

Computational cost is a big factor when working with high-dimensional data. In many cases, lack of time or budget for super computers will dictate what kind of dimensionality techniques are feasible. The authors laid out a table of parameters that influence memory and processing power, which in my opinion is the most useful things presented in this paper. Figure 7 shows the order of certain parameters like original dimensionality *D*, number of instances n, *k* nearest neighbors, target dimensionality *d*, iterations *i* and sparseness factor *p [1]*.

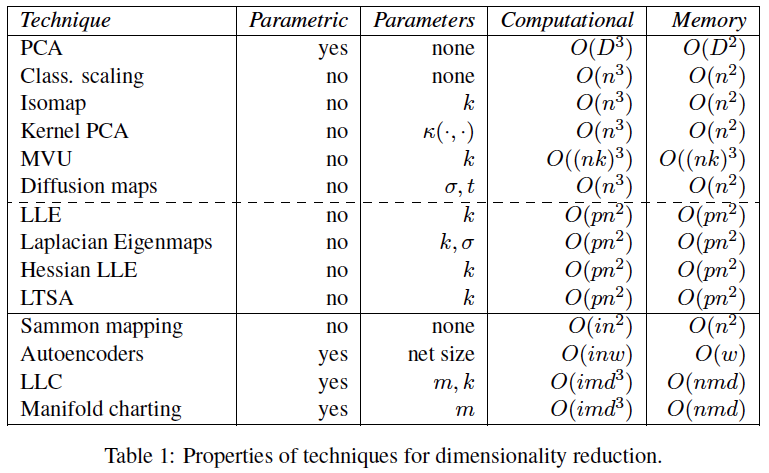


Figure 7

From the table, it is clear that traditional PCA, kernel PCA and Isomaps have similar computational costs. If dimensionality *D* is smaller than *n*, which is often the case, PCA is less exhaustive than other non-linear techniques like kernel PCA and Isomaps. Because LLE is a sparse spectral technique, the sparseness factor p is applied to the computational cost and decreases the overall computation and memory, making it less exhaustive than PCA, kernel PCA and Isomaps. On the contrary, Sammon mapping gets an additional *i* factor due to the iteration involved in minimizing the cost function. For this reason, Sammon mapping can be much more computationally expensive than the other methods if the number of iterations *i > n*.

### Performance on experimental datasets

All thirteen dimensionality reduction techniques were tested on both artificial datasets and natural datasets. The artificial datasets, seen in Figure 8, are meant to test the performance of the techniques on a variety of manifold geometries such as isometric to Euclidean space (Swill roll dataset), non-isometric to Euclidean space (Helix dataset), non-convex (Twin peaks dataset) and discontinuous (Broken Swiss roll) [1]. The techniques were tested on five natural datasets, the MNIST data set of handwritten digit images, the COIL20 dataset consisting of object identification images, NiSIS dataset consisting of pedestrian detection imagery, ORL dataset consisting of face recognition images, and HIVA dataset consisting of drug discovery data.

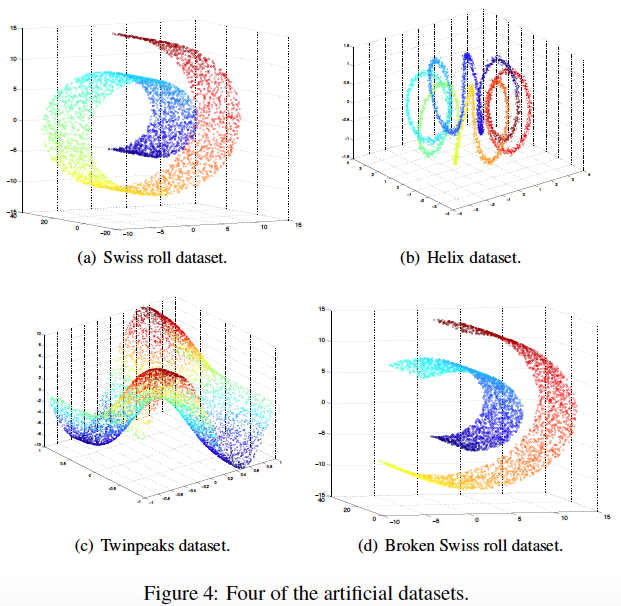


Figure 8

The results of the non-linear techniques on the artificial datasets were impressive. First, the techniques that utilized neighborhood graphs such as Isomaps and LLE outperformed the other techniques including PCA, Kernel PCA and Sammon mapping. Secondly, techniques that aim to preserve the topology of the manifold such as LLE perform well on datasets that are isometric to Euclidean space such as the Swiss Roll, but not well on non-isometric datasets such as the Helix. This is not the case for datasets that use neighborhood graphs. Thirdly, most non-linear techniques did not perform well on the broken swill roll dataset and do not do well with the presence of discontinuities. Lastly, the results show that good performance on the Swill Roll dataset does not generalize to the other three more complex datasets. The comprehensive results from the generalization error can be found in Figure 9 [1]. The bolded values indicate the highest performer for the corresponding dataset.

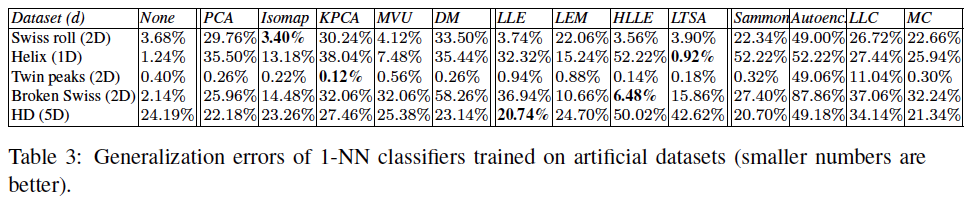


Figure 9

Interestingly, the performance results on the natural datasets were quite the opposite. Specifically, techniques that utilized neighborhood graphs (Isomaps, LLE) performed poorly compared to those that did not (PCA, Kernel PCA, Sammon Mapping). Traditional PCA and the autoencoder technique, which was not discussed in this literature review, outperform the other techniques on four of the five natural datasets. This is contrary to logical thought, that non-linear techniques will perform better than linear techniques on non-linear data. A complete table of the generalization error results is shown in Figure 10 [1].

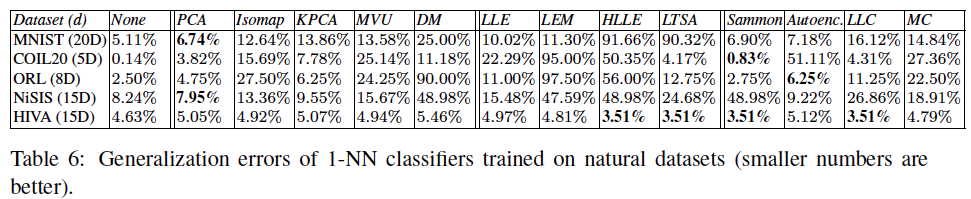


Figure 10

## Conclusion

In conclusion, Maaten et al found that “most non-linear dimensionality reduction techniques do not outperform PCA on natural datasets, despite their ability to learn the structure of complex manifolds” [1]. These non-linear techniques perform strongly on idealized, well sampled and smooth manifolds, but this performance does not always relate to real-world data. The authors summarize three main weaknesses that attribute to the disappointing performance of non-linear techniques on natural datasets. First, the authors have found many flaws in their objective functions. Second, there are numerical problems in the eigen-decomposition such as trivial solutions. Thirdly and least useful, the authors attribute non-linear technique failures to their inherent susceptibility to the curse of dimensionality. The authors suggest that further studies should be done to address each of these weaknesses. The strong performance of autoencoders on natural datasets leads the authors to believe that the most important future of dimensionality reduction research should be using deep learning architecture that use more than one layer of non-linearity.

## References

[1] L. van der Maaten, E. Postma, and J. van den Herik, “Dimensionality Reduction A Comparative Review,” TiCC, TIlburg University, Tilburg, The Netherlands, 2009.

[2] M. Balasubramanian, “The Isomap Algorithm and Topological Stability,” *Science*, vol. 295, no. 5552, pp. 7a – 7, Jan. 2002.

[3] D. Gonsalves and Y. Lyu, “Isomaps: Isometric Feature Mapping,” 01-Sep-2017.

[4] “mlpy v3.2 documentation,” 2014. [Online]. Available: http://mlpy.sourceforge.net/docs/3.2/dim\_red.html. [Accessed: 12-Nov-2018].

[5] R. Lei, “Word Press.” [Online]. Available: https://ryanlei.wordpress.com/2011/04/05/ammai\_07-nonlinear-dimensionality-reduction-by-locally-linear-embedding/.

[6] “Metric MDS,” *Github*, 02-Jan-2018. [Online]. Available: https://jlmelville.github.io/smallvis/mmds.html.