IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 19, 2021, accepted March 5, 2021, date of publication March 17, 2021, date of current version March 24, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3066259

The Impact of Supervised Manifold Learning on Structure Preserving and Classification Error: A Theoretical Study

LAURETA HAJDERANJ, DAQING CHEN, (Member, IEEE), AND ISAKH WEHELIYE
School of Engineering, London South Bank University, SE1 0AA London, U.K.

Corresponding author: Laureta Hajderanj (hajderal@lsbu.ac.uk)

This work was supported under a joint scholarship by the London South Bank University and Acctive Systems, Ltd.

ABSTRACT In recent years, a variety of supervised manifold learning techniques have been proposed to outperform their unsupervised alternative versions in terms of classification accuracy and data structure capturing. Some dissimilarity measures have been used in these techniques to guide the dimensionality reduction process. Their good performance was empirically demonstrated; however, the relevant analysis is still missing. This paper contributes to a theoretical analysis on a) how dissimilarity measures affect maintaining manifold neighbourhood structure and b) how supervised manifold learning techniques could contribute to the reduction of classification error. This paper also provides a cross-comparison between supervised and unsupervised manifold learning approaches in terms of structure capturing using Kendall's Tau coefficients and co-ranking matrices. Four different metrics (including three dissimilarity measures and Euclidean distance) have been considered along with manifold learning methods such as Isomap, *t*-Stochastic Neighbour Embedding (*t*-SNE), and Laplacian Eigenmaps (LE), in two datasets: Breast Cancer and Swiss-Roll. This paper concludes that although the dissimilarity measures used in the manifold learning techniques can reduce classification error, they do not learn well or preserve the structure of the hidden manifold in the high dimensional space, but instead, they destroy the structure of the data. Based on the findings of this paper, it is advisable to use supervised manifold learning techniques as a pre-processing step in classification. In addition, it is not advisable to apply supervised manifold learning for visualization purposes since the two-dimensional representation using supervised manifold learning does not improve the preservation of data structure.

INDEX TERMS Classification error, structure capturing, manifold learning, supervised manifold learning, visualization.

I. INTRODUCTION

Manifold learning is a group of algorithms that seek to learn low dimensional representation embedded in a high dimensional space data. Linear manifold learning techniques such as Principal Component Analysis (PCA) [1] and Multidimensional Scaling (MDS) [2] assume that the low dimensional representation lies in linear manifold(s), and as a result, linear manifold learning methods can be successfully applied to linear data.[1] Conversely, nonlinear manifold learning techniques seek to learn the nonlinear manifold(s) in high dimensional space data [3]. There are some widely used nonlinear manifold learning techniques including Isomap [4], Local Linear Embedding (LLE) [5], Laplacian Eigenmaps (LE) [6], Hessian Eigenmap [7], [8], Local Tangent Space Analysis (LTSA) [9], Maximum Variance Unfolding (MVU) [10], Diffusion Map [11], [12], *t*-Stochastic Neighbour Embedding (*t*-SNE) [13], Topologically Constrained Isometric Embedding [14], Local Coordinates Alignment (LCA) [15], and Uniform Manifold Approximation and Projection (UMAP) [16].

In general, linear manifold learning methods aim to maintain the global structure of the data [3] (far away (close) high dimensional space data samples to be located far away (close)

The associate editor coordinating the review of this manuscript and approving it for publication was Xin Luo.

[1]Linear/nonlinear data are called data in which their low dimensional representation locates on linear/nonlinear manifold(s).

in a low dimensional representation). Conversely, nonlinear manifold learning methods seek to preserve the local structure of data [3]; however, the maintained data structure of the above-mentioned methods depends on the number of neighbours considered [17]. Subsequently, tuning the number of neighbours has a crucial impact on the data structure maintained.

Manifold learning methods have been commonly applied in different fields, including medical images [18], [19] and financial markets [20], to visualize high dimensional data or as a pre-processing step of classification. However, the main focus of manifold learning techniques is on preserving data structure; thus, they may not be useful in classification. Accordingly, Geng *et al.* [21], Hajderanj *et al.* [22], Vlachos *et al.* [23], and Wei *et al.* [24] have proposed supervised manifold learning techniques that use dissimilarity measures[2] to improve the classification accuracy. Furthermore, they have used supervised manifold learning techniques to improve the data structure preservation of their unsupervised versions. The experimental findings in [21]–[25] have illustrated the effectiveness of supervised manifold learning techniques in gaining a better classification model and capturing the data structure more accurately. However, these studies lack theoretical analysis on how a dissimilarity measure affects the classification error and the preservation of manifold neighbourhood structure.

This paper aims to provide a theoretical analysis of the impact of dissimilarity measure on manifold learning methods regarding the preservation of data structure and classification performance. In addition to a theoretical analysis, structure preservation is assessed by Kendall's Tau coefficient and co-ranking matrix. As follows, this paper contributes: 1) to prove that the considered dissimilarity measures could decrease the classification error (radial basis function (RBF)-based classifiers), and 2) to analyze theoretically and to demonstrate experimentally that supervised dimensionality reduction could worse the visualization of high dimensional data in a low dimensional space in terms of structure capturing.

In this paper, a high dimensional data $X^{N\times D}$ is considered with N observations and D features (dimensions), and $Y^{N\times d}$ is considered the low dimensional represenation (manifold) with N samples and d features, where $d \ll D$. x_i and y_i represents the i^{th} data samples in the high and low dimensional spaces, respectively, and l_i represents the i^{th} observation of the class variable L. $dis(a, b)$ signify the Euclidean distance between data samples a and b.

The remainder of this paper is organized as follows: Section II presents a brief review of supervised and unsupervised manifold learning techniques. Section III illustrates the impact of dissimilarity measures on structure capturing, and Section IV presents some experimental results. Section V and Section VI provide the impacts of dissimilarity measures on classification error and some concluded remarks, respectively.

[2]Dissimilarity measures are called metrics that include class information to calculate the similarity between data samples.

II. MANIFOLD LEARNING METHODS

Manifold learning is a group of algorithms that aim to recover the manifold lied in a high dimensional space data. In manifold learning, a low dimensional representation, which lies in a high dimensional space data, is assumed to be a linear or a nonlinear manifold. A linear manifold can be imagined as a plane, whereas a nonlinear manifold can be conceived as a sphere or torus. PCA and MDS are linear manifold learning techniques that assume that the low dimensional representation has a linear shape. In contrast, nonlinear manifold learning techniques, such as Isomap, LLE, LE, *t*-SNE, and UMAP, assume that the low dimensional representation is embedded in nonlinear manifold(s). A brief review of the manifold learning techniques is provided below.

A. PRINCIPAL COMPONENT ANALYSIS (PCA)

PCA is a linear manifold learning method that intends to find a linear projection M of data X to maximize the cost function (1)

$$\max trace(M^T cov(X)M),\ subject\ to\ MM^T = I \qquad (1)$$

where $cov(X)$ is the covariance matrix of the dataset X, and I is an identity squared matrix with 1s in the main diagonal and 0s elsewhere. The low dimensional representation describes the variance of high dimensional space data X, in which the highest variance is represented by the first principal component. PCA fails to perform well in nonlinear data. Furthermore, PCA tends to capture the global data structure, and as a result, it may ignore some local information that may be useful for classification [26].

B. MULTIDIMENSIONAL SCALING (MDS)

MDS is another linear manifold learning technique, which utilizes the cost function (2)

$$\min \sqrt{\sum_{i,j}(dis(x_i, x_j) - dis(y_i, y_j))^2} \qquad (2)$$

to learn the low dimensional representation lied in a high dimensional space data. The main steps of MDS are as follows:

1) Compute the pairwise Euclidean distance matrix D.
2) Convert the distance matrix D to a kernel matrix K by $K = -\frac{1}{2}HDH$, where $H = I - \frac{1}{n}ee^T$ and e is a columns vector of 1.
3) Compute the spectral decomposition of K : $K = UAU^T$, where A is the diagonal matrix and diagonal values are eigenvalues of X^TX, and U is the matrix of eigen vectors of X^TX.
4) Form A^+ by setting $[A^+]_{ij} = \max\{A_{ij}, 0\}$.
5) Set $Y = \sqrt{A^+}U$.
6) Return $[Y]_{n\times d}$.

Like PCA, MDS also does not perform well in maintaining the structure of nonlinear data. Furthermore, MDS favours preserving global data structure because its cost function relates to the pairwise distances, in which large distances have more impact than small ones.

C. ISOMAP

Isomap is a method similar to MDS, but it employs Geodesic distance (*Geo*) instead of Euclidean distance. The pseudocode of Isomap is shown as below:

1) Construct the k-nearest neighbour graph using Euclidean distance.
2) Use Dijkstra's or Floyd's algorithms to calculate the shortest path distances between all data samples, square distances and then store in *Geo*.
3) Apply MDS algorithm with the distance *Geo* as calculated above.

Since Isomap uses Geodesic distance to compute the distance between high dimensional space data samples, it minimizes the following cost function (3):

$$\min \sqrt{\sum_{i,j} (\rho Geo(x_i, x_j) - \rho dis(y_i, y_j))^2} \tag{3}$$

where ρ is the parameter defined as $\rho(D) = -HSH/2$, $S_{ij} = dis(x_i, x_j)^2$, and H is the centring matrix. Isomap is considered a global method due to the approximation of the Geodesic distances *Geo*, which refers to the distance measure that preserves the global geometry of the nonlinear manifold (s) embedded in a high dimensional space data [3].

D. LOCAL LINEAR EMBEDDING (LLE)

LLE is a nonlinear method that reconstructs every data sample as a linear combination of its nearest neighbours. The main steps of LLE are shown as following:

1) Calculate the nearest neighbours based on Euclidean distance.
2) Calculate the reconstruction error as shown below: $\min \sum_{i=1}^{N} \left\| x_i - \sum_{j=1}^{N} w_{ij} x_j \right\|^2$.
3) Compute the low dimensional data Y that best preserves the local geometry, represented by the reconstruction weights.

The low dimensional representation is calculated using the cost function (4).

$$\min \sum_{i=1}^{N} \left\| y_i - \sum_{j=1}^{N} w_{ij} y_j \right\|^2 \tag{4}$$

Because LLE requires that every sample and its neighbours lie on a linear manifold; subsequently, it favours the preservation of the local data structure.

E. LAPLACIAN EIGENMAPS (LE)

LE favours local data structure by calculating the similarity between data samples x_i and x_j, and weight them by providing higher values for close data samples and low values for far away data samples. The main steps of LE are as follows:

1) Nearest neighbour search using Euclidean distance.
2) Define weighted matrix

$$w_{ij} = \left\{ \begin{array}{c} exp(-\frac{dis(x_i,x_j)^2)}{2\sigma^2}) \;\; if \;\; x_j \in Neig_i \\ 0 \;\; otherwise \end{array} \right\}.$$

3) Define with $Neig_i^k$ the neighbourhood of x_i with k neighbours, $D = (d_{ij})$ is a $N \times N$ diagonal matrix with elements $d_{ii} = \sum_{i \in N_i} w_{ij}$, and with $L = D - W$ the graph Laplacian matrix.

The low dimensional representation Y calculates by minimizing the cost function (5)

$$arg \min trace(YLY^T) \tag{5}$$

where $\sum_i \sum_j w_{ij} dis(y_i, y_j) = YLY^T$.

F. T-STOCHASTIC NEIGHBOUR EMBEDDING (T-SNE)

t-SNE favours local structure preservation by weighting the pairwise Euclidean distances in the high dimensional space data using Gaussian distribution, and in the low dimensional space data using Student -t distribution. The main steps t-SNE are concluded as below:

1) Calculate paiwise Euclidean distances $dis(x_i, x_j)$, for $i, j := 1 : N$.
2) Calculate $p_{ij} = \frac{p_{i|j} + p_{j|i}}{2N}$, where $p_{i|j} = \frac{exp(\frac{-dis(x_i,x_j)^2}{2\sigma_i^2})}{\sum_{k \neq i} exp(\frac{-dis(x_i,x_k)^2}{2\sigma_i^2})}$ is the conditional probability between data samples x_i and x_j using Gaussian distribution with variance σ.
3) Calculate $q_{ij} = \frac{(1+dis(y_i,y_j)^2)^{-1}}{\sum_{k \neq l}(1+dis(y_i,y_j)^2)^{-1}}$ between data samples y_i and y_j in the low dimensional space using Student t-distribution with degree of freedom 1.

t-SNE minimizes the similarity of the high and low dimensionality space data using Kullback-Leibler cost function (6).

$$\min \sum_{i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}} \tag{6}$$

The similarity p_{ij} is large for close data samples and smoothly decreases as the distance becomes greater; thus, t-SNE favours data local structure capturing.

G. UNIFORM MANIFOLD APPROXIMATION AND PROJECTION (UMAP)

UMAP, which is a method similar to t-SNE, has the following main steps:

1) Calculate paiwise Euclidean distances $dis(x_i, x_j)$, for $i, j := 1 : N$.
2) Calculate similarities $v_{ij} = (v_{i|j} + v_{j|i}) - v_{i|j} V_{j|i}$, where $v_{i|j} = exp(-dis(x_i, x_j) - \rho)/\sigma_i]$, and ρ and σ are defined as below:

$\rho_i = \min(dis(x_i, x_j), 1 \le j \le k, dis(x_i, x_j) \ge 0) \sum_{j=1}^{k} exp(\frac{-\max 0, dis(x_i,x_j)-\rho_i}{\sigma_i}) = \log_2 k$. The value $v_{ij}^{\cdot}$ is calculated as $v_{ij}^{\cdot} = (1 + a||y_i - y_j||_2^{2b})^{-1}$, where a are b positive parameters defined by the user.

The low dimensional representation is corresponding Y that minimizes the cost function (7)

$$\min \sum_{i \neq j} v_{ij} \log \frac{v_{ij}}{v_{ij}^{\cdot}} + (1 - v_{ij}) \log \frac{(1 - v_{ij})}{(1 - v_{ij}^{\cdot})} \tag{7}$$

Like t-SNE, UMAP favours small distance preservation.

In general, the main focus of manifold learning techniques is on preserving data structure; thus, they may not be useful in classification. Their supervised versions have been proposed to improve specific manifold learning techniques in terms of classification accuracy. Furthermore, supervised manifold learning techniques have also been proposed to capture better the high dimensional data structure than their unsupervised versions. But considering that the purpose of this study is to assess how dissimilarity measures affect classification accuracy and the data structure capturing, in the following, we have considered only those supervised manifold learning techniques that use dissimilarity measure instead of Euclidean distance to calculate the similarity between data samples.

H. SUPERVISED MANIFOLD LEARNING

Supervised manifold learning techniques employ dissimilarity measures (dis_1, dis_2, dis_3) instead of Euclidean distance to define the pairwise distance matrix or construct the neighbourhood graph as shown below:

$$dis_1 = \begin{cases} \sqrt{1 - e^{\frac{-dis(x_i,x_j)^2)}{\beta}}} & l_i = l_j \\ \sqrt{e^{\frac{dis(x_i,x_j)^2)}{\beta}} - \alpha} & l_i \neq l_j \end{cases} \tag{8}$$

$$dis_2 = \begin{cases} \frac{1}{\psi} dis(x_i, x_j) & l_i = l_j \\ dis(x_i, x_j) & l_i \neq l_j \end{cases} \tag{9}$$

$$dis_3 = \begin{cases} dis(x_i, x_j) & l_i = l_j \\ dis(x_i, x_j) + \mu \max(dis(x_i, x_j))\lambda_{ij} & l_i \neq l_j \end{cases} \tag{10}$$

The main difference between supervised and unsupervised manifold learning techniques lies in the first step of each algorithm, and the rest of the steps are the same for both versions. Dissimilarity measures enforce the same class data samples to be close and different class data samples to be far away.

A supervised version of Isomap was proposed by Geng *et al.* [21], in which a neighbourhood graph was designed concerning dissimilarities between data samples, and each data sample $x_i \in X$ chooses k neighbours with dissimilarity measure dis_1 less than a given threshold ε. Supervised Isomap was tested in two datasets, Face Images and Swiss Roll [21]. The authors claimed that using dissimilarity measure dis_1 (8) to Isomap enhanced visualization in terms of structure capturing and achieved a more accurate and robust classification model. Dissimilarity measure dis_1 was also applied to calculate the pairwise distances between data samples in supervised t-SNE [22]. The difference between t-SNE and supervised t-SNE is that in supervised t-SNE, the pairwise distance is calculated using dis_1 instead of Euclidean distance that t-SNE uses. Supervised t-SNE was tested in datasets such as MNIST [27], SEER Breast Cancer [28] and Chest X-ray [29] and achieved lower classification error compared with unsupervised t-SNE. Dissimilarity measure dis_1 was also implemented to construct the neighbourhood graph at the first step of LLE (ESLLE [30]) to achieve a higher classification accuracy in Swiss Roll data.

Other dissimilarity measures such as dis_2 and dis_3 have been implemented to Isomap (WeightedIso [23]) and LLE(SLLE [25]). WeightedIso was implemented in datasets Iris, Liver, Lung Sonar, Glass, and Image, to achieve lower classification error. SLLE calculated the neighbourhood graph using dis_3, where $\mu \in [0, 1]$ and λ_{ij} is 0 if data samples i and j are from the same class, and 1 otherwise. Yu *et al.* [31] and Cheng *et al.* [32] proposed supervised versions of t-SNE, where the distance between different classes data samples is defined in (11)

$$dis_4 = \begin{cases} dis(x_i, x_j) e^{v(x_i)-v(x_j)} & l_i = l_j \\ dis(x_i, x_j) & l_i \neq l_j \end{cases} \tag{11}$$

where $v_{(x_i)}$ refers to the angle information [31] and the silhouette frame information [32] of the sample x_i. Although it is not a published article, a supervised version of UMAP[3] has also been proposed, with the purpose of capturing the structure of high dimensional space data. Furthermore, a recent preprint article [33] suggests using label information to produce a better visualization in terms of retaining the manifold structure.

Overall, supervised manifold learning techniques have been proposed to improve classification accuracy and improve visualization in terms of data structure preservation. However, there lacks theoretical analysis on the impact of dissimilarity measures on classification error. Furthermore, supervised manifold learning has been widely used to visualize high dimensional data assuming to retain the manifold structure better, which is not true. In the following sections, we provide analysis based on some theoretical foundations to confirm that dissimilarity measures in manifold learning techniques do not help capture the manifold structure better but destroy it. In other words, the use of dissimilarity measures in manifold learning techniques generate low dimensional visualizations that do not represent the real structure of the manifold embedded in the high dimensional space data.

III. THE IMPACT OF DISSIMILARITY MEASURE ON STRUCTURE CAPTURING

In most of the manifold learning techniques, the nearest-neighbour search is the first step, where a distance measure

[3]https://umap-learn.readthedocs.io/en/latest/supervised.html

is employed to find data samples that are neighbours in a manifold [34]. The integrity of a manifold learning technique depends on the goodness of maintaining the neighbourhood structure of the manifold hidden in a high dimensional space data. Preserving the neighbourhood structure means close (far away) data samples of the original space embed close (far away) in a lower dimensional space. Thus, the best manifold learning technique is a method that generates the low dimensional space data that maintains the best neighbourhood structure of high dimensional space data. To better understand which of the distance measures (dissimilarity measures) should be applied, we should first explain the manifold concept.

A manifold M^d, also known as topological manifold, is a topological space that is locally a Euclidean space and a Second Countable space. A Euclidean space is a space with a finite number of dimensions, where coordinates present each data sample (one per each dimension). The distance between any two data samples is calculated using the Pythagorean theorem, where the distance between the data sample a with n coordinates $(a_1, \ldots, a_n)$ and data sample b with n coordinates $(b_1, \ldots, b_n)$ is calculated using $\sqrt{(a_1 - b_1)^2 + \ldots + (a_n - b_n)^2}$, which corresponds to an Euclidean distance.

Unsupervised manifold learning techniques use the Euclidean or Geodesic distance to calculate each data samples nearest neighbours in a manifold. On the other hand, supervised manifold learning techniques employ dissimilarity measure to calculate the nearest neighbours of each data sample. Dissimilarity measures dis_1 (8), dis_2 (9), and dis_3 (10) search the nearest neighbours by forcing the same class data samples to be close and/or forcing the different class data samples to be far away. As a consequence, for a given data sample, different neighbours set may be produced when using various measures such as Euclidean distances (dis), dis_1, dis_2, and dis_3. Each manifold learning technique seeks to keep the neighbourhood structure (neighbours set) defined in the high dimensional space data. Thus, four different low dimensional representations will be generated if four different neighbours sets have been defined in the high dimensional space data. However, the local neighbourhood structure of a manifold is determined using the Euclidean distance because a manifold is conceived to be a locally Euclidean space. A theoretical analysis of the impact of dissimilarity measure on data structure capturing will be illustrated in the following section, which is also supported by a practical demonstration.

A. THEORETICAL ANALYSIS

Let RO be the order set which contains the Euclidean distance of each data sample and its k-nearest neighbours of the high dimensional space data. Based on the manifold definition, Euclidean distance is the metric that calculates the local[4] neighbours for each data sample. Alternatively, we define RO_1, RO_2, and RO_3 as order sets that contain the distances of data samples and their k-nearest neighbours in the high dimensional using the dis_1, dis_2, and dis_3, respectively. We also define ro_1, ro_2, and ro_3 as order sets that contain distances of the low dimensional data samples and their k-nearest neighbours using the dis_1, dis_2, and dis_3, respectively. To simplify our analysis, we consider that every manifold learning approach has perfectly embedded data,[5] and as a result, $ro = RO$, $ro_1 = RO_1$, $ro_2 = RO_2$, and $ro_3 = RO_3$.

A manifold learning technique maintains the manifold structure (the one that is locally Euclidean space) if the order set RO is the same with ro. To determine whether the neighbourhood structure has been captured, we must prove whether dis, dis_1, dis_2, and dis_3 are order isomorphism functions. In accordance with that, we use *Proposition 1*, *Definition 1*, and *Definition 2* that defines a function as order-isomorphism, bijective, and order-preservation, respectively.

Proposition 1 Let I and J be two order sets, then the function $f : I -> J$ is called an order-isomorphism function if f is:

1) bijective, and
2) order-preservation (for all $a, b \in I$ we have $a \leq b \Leftrightarrow f(a) \leq f(b)$).

Definition 1 A bijective function should be: 1) injective and 2) surjective. Let be I and J two sets, then the function $f : I -> J$ is injective if and only if whenever $f(a) = f(b)$ then $a = b$ for $a, b \in I$, and is surjective if and only if for every $d \in J$, there is at least one $c \in I$ such that $f(c) = d$.

Definition 2 Let I and J be two order sets, then the function $f : I -> J$ is called an order-preservation function if for all elements $a, b \in I$, and $f(a), f(b) \in J$, $a \leq b \iff f(a) \leq f(b)$.

Consider $dis : RO -> RO$, $dis_1 : RO -> RO_1$, $dis_2 : RO -> RO_2$, and $dis_3 : RO -> RO_3$. We can re-write functions dis, dis_1, dis_2, and dis_3 as: $dis : dis(x_i, x_j) -> dis(x_i, x_j)$,

$$dis_1 : dis(x_i, x_j) -> \begin{cases} \sqrt{1 - e^{\frac{-dis(x_i,x_j)^2)}{\beta}}} & l_i = l_j \\ \sqrt{e^{\frac{dis(x_i,x_j)^2)}{\beta}}} - \alpha & l_i \neq l_j, \end{cases}$$

$$dis_2 : dis(x_i, x_j) -> \begin{cases} \frac{1}{\psi} dis(x_i, x_j) & l_i = l_j \\ dis(x_i, x_j) & l_i \neq l_j, \end{cases}$$

$$dis_3 : dis(x_i, x_j) -> \begin{cases} dis(x_i, x_j) \\ \quad l_i = l_j \\ dis(x_i, x_j) + \max(dis(x_i, x_j))\mu \\ \quad l_i \neq l_j. \end{cases}$$

Proposition 2 dis is an order-isomorphism function whereas, dis_1, dis_2, and dis_3 are not order-isomorphism functions.

[4]Define with local k-nearest neighbours.

[5]The manifold learning loss function has achieved its optimal value (zero).

Proof: Based on Proposition 1, a function is order-isomorphism if it is: 1) bijective and 2) order-preservation. To check if dis, dis_1, dis_2, and dis_3 are order-isomorphism functions, we firstly have to check if they are bijective and order-preservation functions.

The first condition checks whether dis, dis_1, dis_2, and dis_3 are bijective functions.

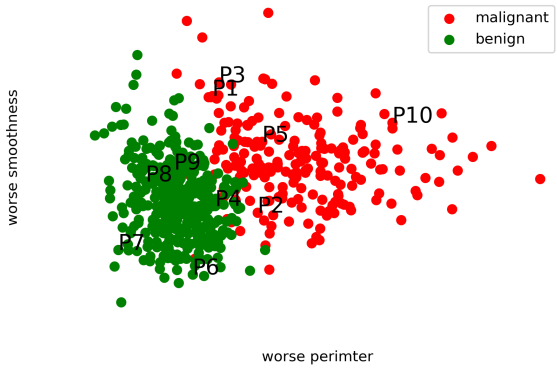
1) dis is a bijective, because it is injective and surjective. Suppose $a = dis(x_1, x_2), l(x_1) = l(x_2)$, $b = dis(x_1, x_3), l(x_1) \neq l(x_3)$, and $a = b = 2$. Since $dis : dis(x_i, x_j) -> dis(x_i, x_j)$, then $dis(dis(x_1, x_2)) = dis(x_1, x_2) = 2$, and $dis(dis(x_1, x_3)) = dis(x_1, x_3) = 2 \Leftrightarrow dis(dis(x_1, x_2)) = dis(dis(x_1, x_3)) \Rightarrow dis$ is an injective function. dis is also surjective, because $dis(dis(x_i, x_j)) = dis(x_i, x_j) \Leftrightarrow$ for every $dis(x_i, x_j)$, there exist at least one $dis(x_i, x_j)$ that $dis(dis(x_i, x_j)) = dis(x_i, x_j)$.
2) The function $dis : RO -> RO$ is an order-preservation function because the identity map is an order-preservation function.

In conclusion, dis is a bijective and an order-preservation function; thus, it is an order-isomorphism function.

Let check if dis_1 is bijective and order-preservation function.

1) Let $a = dis(x_1, x_2), l(x_1) = l(x_2)$, $b = dis(x_1, x_3), l(x_1) \neq l(x_3)$, where $a = b = 2$. We can prove that $dis_1(a) \neq dis_1(b)$. Let consider $\beta = 1$ and $\alpha = 0.5$, then we have $dis_1(dis(x_1, x_2)) = \sqrt{1 - e^{\frac{-2^2}{1}}} = 0.9908$ and $dis_1(dis(x_1, x_3)) = \sqrt{e^{\frac{2^2}{1}} - 0.5} = 6.8890$. As a result $dis_1(dis(x_1, x_2)) \neq dis_1(dis(x_1, x_3))$.
2) To check if dis_1 is an order-preservation function, the order-preservation condition between each two order sets RO and RO_1 must be satisfied. Suppose $RO = \{dis(x_1, x_2), dis(x_1, x_3)\}$ and $RO_1 = \{dis_1(x_1, x_2), dis_1(x_1, x_3)\}$, where $dis(x_1, x_2) = 4$, and $dis(x_1, x_3) = 4.1$, thus, $RO = \{4, 4.1\}$. Conversely, x_1 and x_2 have different classes, and as a result, data samples x_1 and x_2 have been enforced to be far away with $dis_1(x_1, x_2) = 13.8919$. By contrast, data samples x_1 and x_3, which have the same class, have been enforced to be closer with $dis_1(x_1, x_3) = 0.9975$ for $\alpha = 0.5$, and as a conclusion, dis_1 is not an order-preservation function.

Since dis_1 is not injective function, it is not bijective function. Furthermore, dis_1 is not order-preservation function; as a conclusion it is not order-isomorphism function. Like dis_1, dis_2 and dis_3 are not bijective and order-preservation functions. Note that dis_2 favours the same class neighbours by decreasing their Euclidean distance with a positive value ψ. On the other hand, dis_3 favours the same class data samples by increasing the distance between data samples from different classes. As a result, the local manifold structures defined by dis_2 and dis_3 are not the same as the manifold structure defined by Euclidean distance, which is the distance that a manifold is assumed to use.



(a)

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
P1 rank index	1	5	3	2	5	3	5	5	4	6
P2 rank index	6	1	5	6	2	7	8	8	8	3
P3 rank index	3	3	1	3	3	5	7	7	6	4
P4 rank index	2	4	2	1	4	4	6	6	5	5
P5 rank index	7	2	6	7	1	8	9	9	9	2
P6 rank index	4	6	4	4	6	1	4	4	2	7
P7 rank index	9	10	9	9	10	9	1	2	7	10
P8 rank index	8	8	8	8	8	6	2	1	3	9
P9 rank index	5	7	7	5	7	2	3	3	1	8
P10 rank index	10	9	10	10	9	10	10	10	10	1

(b)

FIGURE 1. The visualisation of *worse perimeter* and *worse smoothness* variables from Breast Cancer dataset (a), and the neighbourhood rank indexes between ten randomly selected patients (b).

Overall, dis is a bijective and an order-preservation function, such that it is an order-isomorphism function. On the other hand, dis_1, dis_2, and dis_3 are neither bijective functions or order-preservation functions, and subsequently they are not order-isomorphism functions. Thus, the low dimensional visualization produced by a manifold learning using dissimilarity measure is not the best representation of the high dimensional data structure. □

To better understand the impact of dissimilarity measure on manifold learning techniques in terms of structure capturing, we apply Breast Cancer data in Isomap (uses Euclidean distance) and Supervised Isomap (uses dis_1), illustrated in the next subsection.

B. PRACTICAL ANALYSIS

Breast Cancer data has been selected to demonstrate practically the impact of dissimilarity measure on structure capturing. To simplify the demonstration, we have considered two variables *worse perimeter* and *worse smoothness* of the Breast Cancer data[6] and then have been considered ten

[6]Breast Cancer with 569 samples and 30 variables from Sklearn, Python.

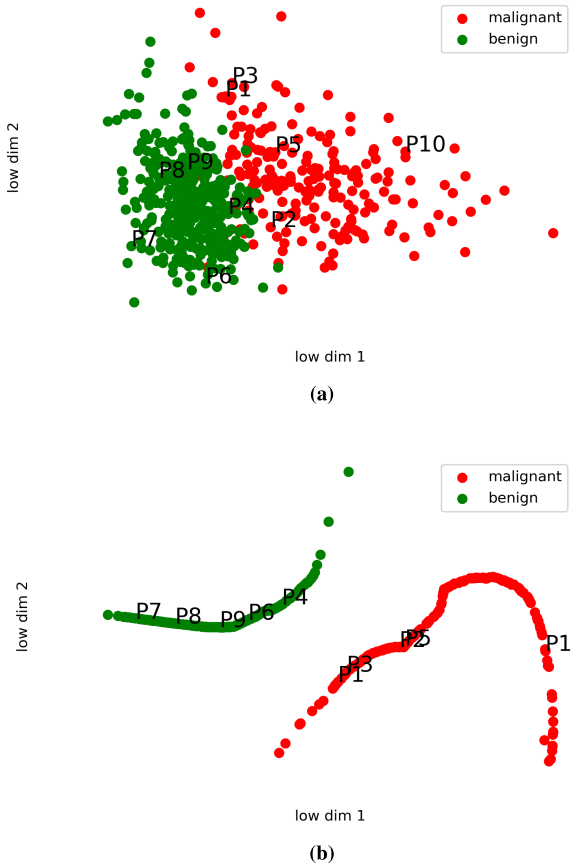


FIGURE 2. The visualization of low dimensional representation of Isomap (a) and the visualization of low dimensional representation of Supervised Isomap.

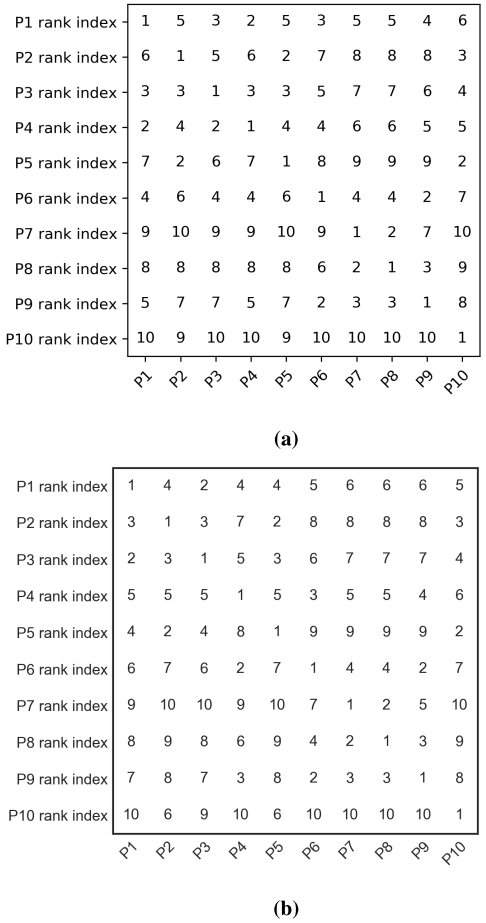


FIGURE 3. The neighbourhood rank indexes of the low dimensional space data generated by Isomap (a), and Supervised Isomap (b).

randomly-selected data samples. Each data sample corresponds to a patient, and we have built the neighbours rank indexes[7] for each selected patient, as shown in Fig. 1. Considering patient 1; the nearest neighbour of patient 1 is patient 4 (rank 2), followed by patient 3 (rank 3), patient 6 (rank 4), patient 9 (rank 5), patient 2 (rank 6), patient 5 (rank 7), patient 8 (rank 8), patient 7 (rank 9), and patient 10 (rank 10).

To evaluate which of the methods has retained the data structure better, we have constructed a difference matrix named *Retained-Structure* that contains the difference between the neighbourhood rank matrix of the high and the low dimensional space data. In an ideal case, the Retained-Structure matrix contains only element 0 (zero). Nonzero elements, which indicate a failure in retaining the neighbourhood structure, are positive or negative numbers. A positive number $P_{ij} = +in$ indicates that the method has jumped $+in$ positions closer the j^{th} data sample to the i^{th} data sample. By contrast, a negative number $P_{ij} = -in$ indicates that the method has been forced the i^{th} data sample to be in positions further away from the j^{th} data sample. In terms of medical interpretation, we can say that *worse perimeter* and *worse smoothness* variables of patient 1 are the most similar to patient 4 and the least similar to patient 10. Thus, if applying any manifold learning technique to the above-considered data, the best manifold learning (dimensionality reduction) method is the one that maintains the neighbourhood structure. In other words, patient 1 should maintain the neighbours rank in the following order: patient 4, patient 3, patient 6, patient 9, patient 2, patient 5, patient 8, patient 7, and patient 10 from the closest to the most distant patient.

To demonstrate the impact of a dissimilarity measure on structure capturing, we apply dis_1 to Isomap and have compared with the standard Isomap. Visually, supervised Isomap with dis_1 seems better, as samples of the same class are closer, and samples of different classes have become more separated. However, the visualization of standard Isomap seems more similar to the visualization of the original data, which is discussed below.

The Retained-Structure matrices generated by Isomap and Supervised Isomap are showed in Fig. 3(a) and Fig. 3(b), respectively. Fig. 3 shows that the method that has captured the neighbourhood structure entirely is Isomap, as its Retained-Structure matrix Fig. 4(a) contains only elements 0.

[7]The neighbourhood ranking index demonstrates the neighbourhood ranking index among patients.

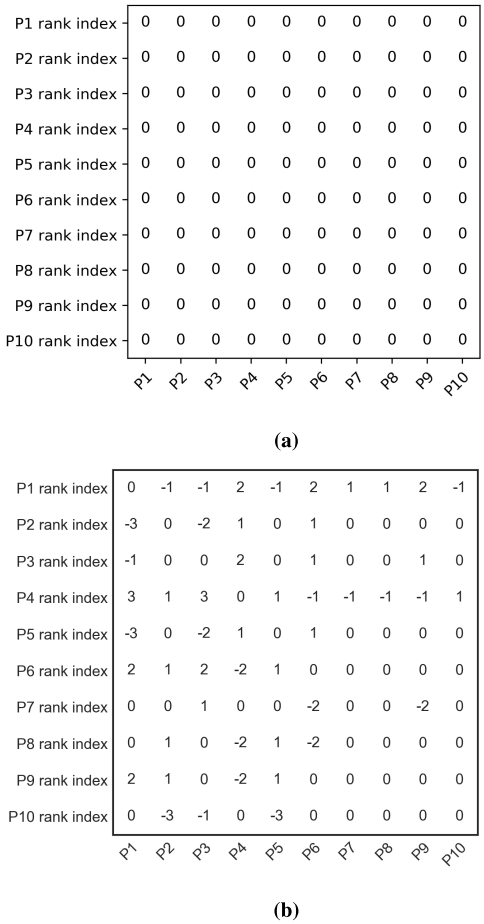


FIGURE 4. Retained-Structure matrix of Isomap (a), and Supervised Isomap (b).

Contrastingly, the supervised Isomap has failed to maintain the neighbourhood structure, demonstrated by nonzero elements in the Retained-Structure matrix Fig. 4(b). Patients are organized into two classes where patient 1, patient 2, patient 3, patient 5, and patient 10 are patients diagnosed with *malignant*, whereas patient 4, patient 6, patient 7, patient 8, and patient 9 are patients diagnosed with *benign*. We can spot from the Retained-Structure matrix Fig. 4(b) that the same class samples have been forced to be closer, demonstrated by negative values in the Retained-Structure matrix, shown in Fig. 4(b). Different class patients have been forced to be further away, illustrated by positive values in the Retained-Structure, matrix shown in Fig. 4(b). We conclude that forcing data samples to be closer or further away impacts the scale of maintaining the neighbourhood structure. As shown in Fig. 2(b), patient 1 was more similar to patient 4 in terms of *worse perimeter* and *worse smoothness* variables. However, using supervised Isomap, the nearest patient to patient 1 is patient 3, shown in Fig. 3(b). Consequently, we can assume that patient 1 and patient 3, which are very close in the visualization of low dimensional representation, may need the same treatment. However, patient 1 and patient 3 have different corresponding values of *worse perimeter* and *worse smoothness* in the original data. As a result, the aforementioned decision for the same treatment may be wrong.

TABLE 1. Kendall's tau for methods (columns) using metrics (rows) in breast cancer data.

		Methods		
		Isomap	*t*-SNE	LE
Metric	Euclidean	0.9977	0.8150	0.7267
	dis_1	0.8288	0.7291	0.3878
	dis_2	0.8528	0.7025	0.0941
	dis_3	0.3192	0.8137	0.0988

IV. EXPERIMENTAL RESULTS AND DISCUSSION

Isomap, *t*-SNE, and LE are three manifold learning techniques considered in this paper. They have been tested with two datasets, Breast Cancer and Swiss Roll, using Euclidean distance and three dissimilarity measures dis_1 (8), dis_2 (9), and dis_3 (10). The selected manifold learning techniques were implemented in Python using the corresponding Sklearn versions and the same number of iterations (2000). Supervised manifold learning methods were also implemented using their Sklearn versions, but by selecting the *pre-computed* metric, where we pre-computed the dissimilarity measures, separately. Their performance in maintaining the neighbourhood structure of data in a manifold has been evaluated by Kendall's Tau coefficients and co-ranking matrices [35]. Furthermore, we have tuned the number of neighbours for each method from 1 to $N-1$, because the number of neighbours considered has a substantial impact on the scale of preserving the neighbourhood structure of a manifold.

A. BREAST CANCER

Breast Cancer data with 569 data samples (patients), thirty variables and two classes is the first dataset considered. The thirty-dimensional data will be transformed to two-dimensional space data (visualization in Fig. 7) by employing four different metrics, such as Euclidean distance, dis_1, dis_2, and dis_3 to Isomap, *t*-SNE, and LE. Their performances have been evaluated by Kendall's Tau coefficients presented in Table 1, and co-ranking matrices demonstrated in Fig. 8. The experiments conducted on Breast Cancer data show that Euclidean distance helps Isomap (k: 515) to capture the best data structure as demonstrated by a nearly diagonal co-ranking matrix demonstrated in Fig. 8(a), and Kendall's Tau coefficient with 0.9977, as shown in Table 1. The dis_1, dis_2, and dis_3 used in Isomap are less useful in capturing the neighbourhood structure, estimated by Kendall's Tau coefficients (Table 1), and the co-ranking matrices (Fig. 8). The Euclidean distance has resulted in the best metric for *t*-SNE, regarding the maintenance of the data structure, with a Kendall's Tau coefficient of 0.8150. However, dis_3 demonstrated excellent performance by competing with Euclidean distance for *t*-SNE. Note that the Gaussian distribution becomes broader because if σ increases and the broader the Gaussian distribution is, the more sensitive it becomes to more distant neighbours. This conclusion is supported by the

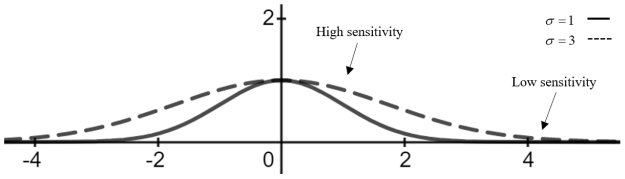


FIGURE 5. Two Gaussian distributions.

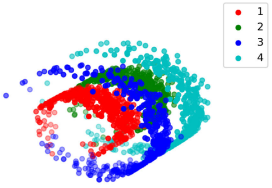


FIGURE 6. Swiss Roll data.

result of the co-ranking matrix of *t*-SNE using dis_3, which has fewer off-diagonal entries. Contrastingly, dissimilarity measure dis_2 enforces the data samples of the same class to have a smaller distance; and as such, the number of data samples with small distances becomes higher. As a result, the Gaussian distribution(s), which relates to the density of data σ, becomes sharp when the density is small. Having a sharp Gaussian distribution means that the distribution is more sensitive at small distances than large ones, as shown in Fig. 5. Thus, the number of entries that are part of the sensitive distribution section is higher, which means an improvement in capturing the local data structure.

LE using the Euclidean distance preserves the data structure better than using other metrics, supported by their co-ranking matrices and Kendall's Tau coefficients. The dissimilarity measure dis_1 employed to LE reduces Kendall's Tau coefficient by 0.3878, as demonstrated in Table 1. The deterioration of the structure preservation can be seen in the respective co-ranking matrices, as shown in Figs. 8 (j), in which the supervised LE has more off-diagonal entries.

B. SWISS ROLL

The second dataset considered is the three-dimensional Swiss Roll data with 1600 data samples (shown in Fig. 6), which will be transformed into two-dimensional space data, by using four different metrics including *dis*, dis_1, dis_2, and dis_3 in Isomap, *t*-SNE, and LE. Performances of Isomap, *t*-SNE, and LE using *dis* (Euclidean distance), dis_1, dis_2, and dis_3 with Swiss Roll data, were estimated using Kendall's Tau coefficients as shown in Table 2, and co-ranking matrices illustrated in Fig. 10. The two-dimensional data visualizations are demonstrated in Fig. 9. Based on Kendall's Tau coefficient values and co-ranking matrices, manifold learning techniques that employ Euclidean distance, have preserved better Swiss Roll data than three other metrics (dis_1, dis_2, and dis_3). Among unsupervised manifold learning methods,

TABLE 2. Kendall's tau for methods (columns) using metrics (rows) in swiss roll data.

		Methods		
		Isomap	*t*-SNE	LE
Metric	Euclidean	0.9121	0.8700	0.9043
	dis_1	0.8269	0.8268	0.8508
	dis_2	0.2473	0.7686	0.7120
	dis_3	0.3192	0.8460	0.8515

Isomap (Euclidean distance) captures the best Swiss Roll data structure, with Kendall's tau 0.9121. The LE with dis_1 captures the best data structure across supervised methods, with Kendall's tau 0.8508.

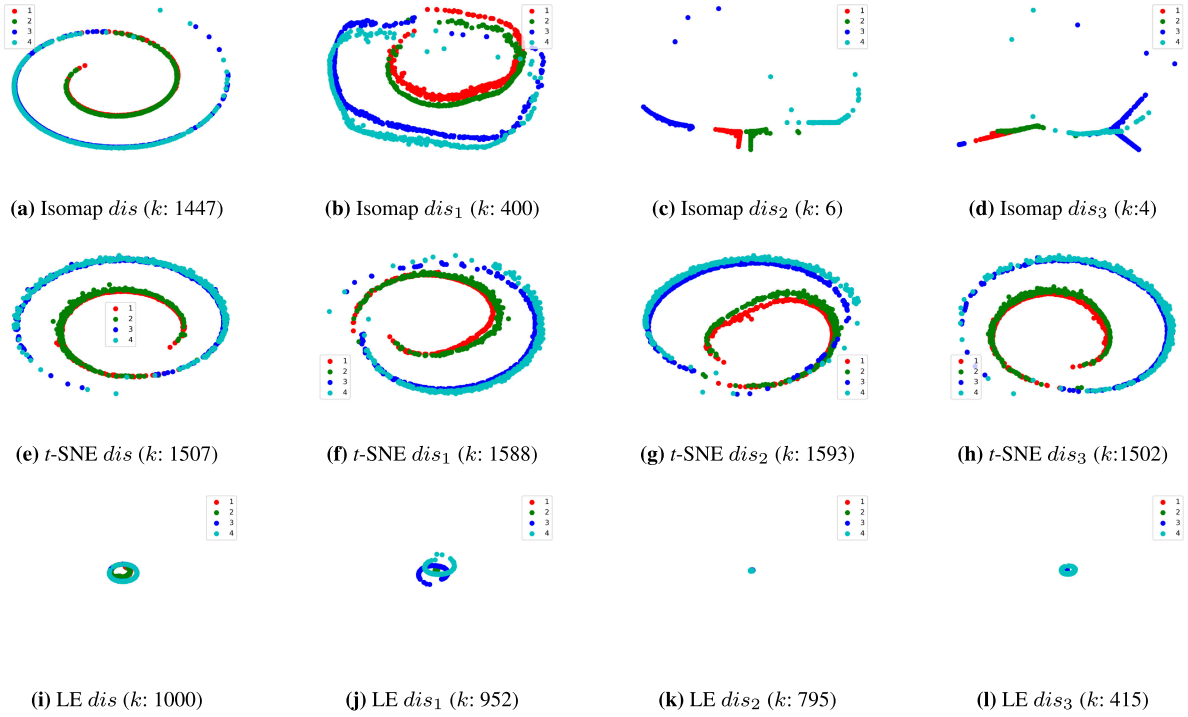
Unlike with Breast Cancer data, in Swiss Roll data *t*-SNE managed to capture the highest data structure by using Euclidean distance and not a dissimilarity measure. However, among dissimilarity measures, dis_3 resulted in capturing global data structure the best (*t*-SNE shown in Fig. 10(h)). As previously noted, the broader the distance range of data, the broader the Gaussian distribution and the more sensitive to large distances it is, the more it improves the data structure capturing.

Overall, employing a dissimilarity measure in a manifold learning technique does not improve data structure preservation. However, in some scenarios, dis_3 helps *t*-SNE to capture a more global data structure, but it may lose some local information.

V. THE IMPACT OF DISSIMILARITY MEASURE ON CLASSIFICATION PERFORMANCE

A manifold learning technique can be employed as a pre-processing step for classification. However, the priority of a manifold learning technique is to capture data structure instead of separate data samples of different classes. Consequently, researchers have proposed class information in calculating the similarity between data samples (dissimilarity measures), i.e., dis_1, dis_2, and dis_3, in manifold learning to achieve a lower classification error. This section discusses how the dissimilarity measure affects a classification model to achieve a lower classification error.

Consider a manifold learning M that generates low dimensional data Y, Y_1, Y_2, and Y_3 using metrices *dis* (Euclidean distance), dis_1, dis_2, and dis_3, respectively. To simplify our analysis, we consider that the manifold learning method M has performed perfectly (the loss function employed in the manifold learning has reached i.e., its minimal value (zero)), such that the neighbourhood structures defined in the high dimensional space using Euclidean distance (*dis*), dis_1, dis_2, and dis_3 are preserved completely. Note that the neighbourhood structures defined using *dis*, dis_1, dis_2, and dis_3 are the same with the neighbourhood structure defined using *dis* in the low dimensional data Y, Y_1, Y_2, and Y_3, respectively. Our theoretical analysis is based on the work of Balcan *et al.* [36] who proposed the (ϵ, γ) good similarity function based on intuitive and sufficient conditions that allow a similarity function to learn well, supported by *Definition 3*, *Definition 4*, *Theorem 1*, and *Theorem 2*.

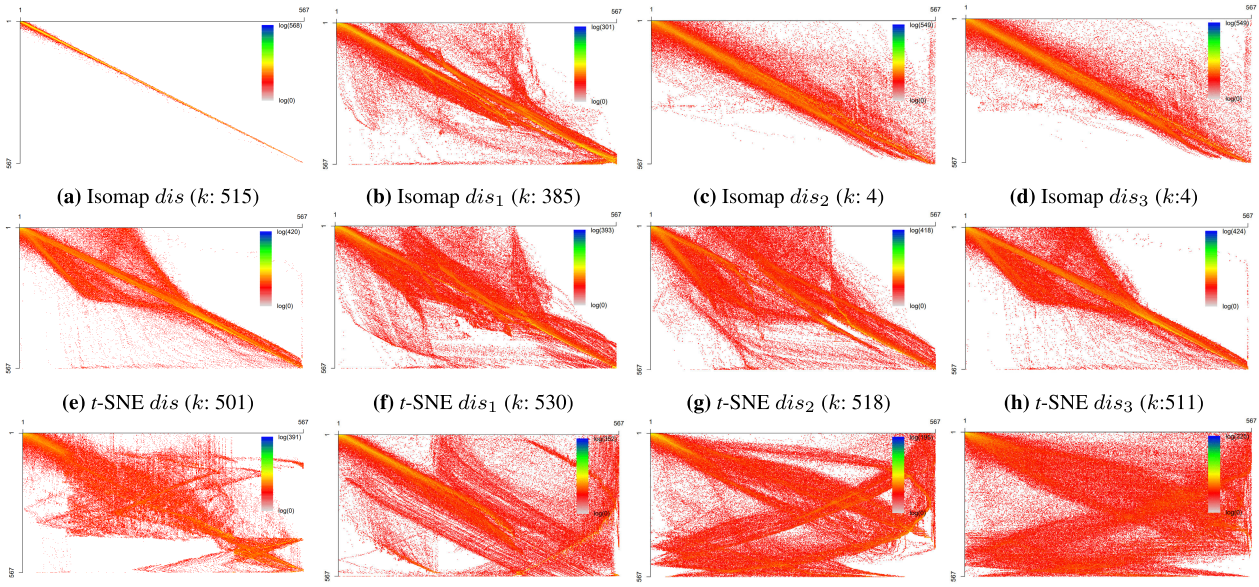


(a) Isomap dis (k: 1447) **(b)** Isomap dis_1 (k: 400) **(c)** Isomap dis_2 (k: 6) **(d)** Isomap dis_3 (k:4)

(e) t-SNE dis (k: 1507) **(f)** t-SNE dis_1 (k: 1588) **(g)** t-SNE dis_2 (k: 1593) **(h)** t-SNE dis_3 (k:1502)

(i) LE dis (k: 1000) **(j)** LE dis_1 (k: 952) **(k)** LE dis_2 (k: 795) **(l)** LE dis_3 (k: 415)

FIGURE 7. Visualization of two-dimensional breast cancer data generated by ISOMAP, *t*-SNE and LE using as metric euclidean distance, DIS_1, DIS_2, and DIS_3.



(a) Isomap dis (k: 515) **(b)** Isomap dis_1 (k: 385) **(c)** Isomap dis_2 (k: 4) **(d)** Isomap dis_3 (k:4)

(e) t-SNE dis (k: 501) **(f)** t-SNE dis_1 (k: 530) **(g)** t-SNE dis_2 (k: 518) **(h)** t-SNE dis_3 (k:511)

(i) LE dis (k: 426) **(j)** LE dis_1 (k: 169) **(k)** LE dis_2 (k:189) **(l)** LE dis_3 (k: 285)

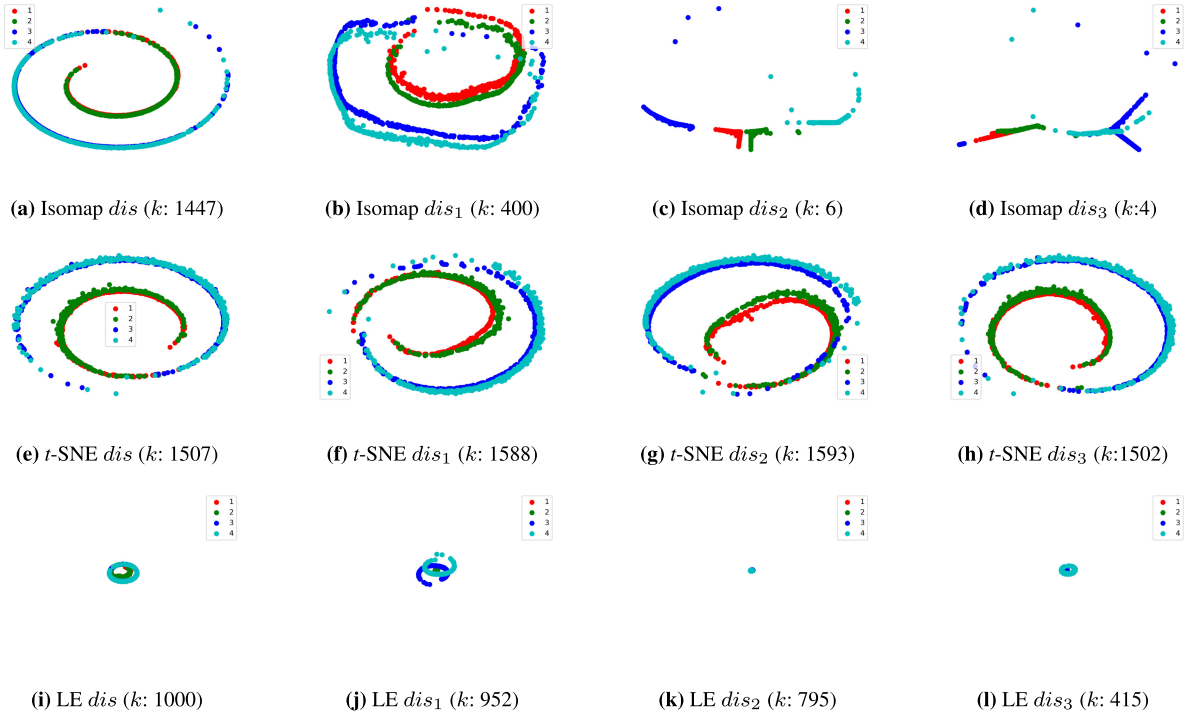
FIGURE 8. Co-ranking matrixes of two-dimensional breast cancer data generated BY ISOMAP, *t*-SNE and LE using as metric euclidean distance, DIS_1, DIS_2, and DIS_3.

Definition 3 (Balcan *et al.* [36]) A similarity function over Y is any pairwise function $K : X \times X- > [-1, 1]$.

Definition 4 (Balcan *et al.* [36]) K is a strongly (ϵ, γ) good similarity function, if at least a- $(1 - \epsilon)$ probability mass of examples y satisfy: $E_{y-Y}[dis(y, y')|l(y') \neq l(y)] > E_{y-Y}[dis(y, y')|l(y') = l(y)] + \gamma$.

Theorem 1 (Balcan *et al.* [36]) *If K is a valid kernel function, and is (ϵ, γ)-good similarity for some learning problem, then it is also (ϵ, γ)-kernel-good for the learning problem.*

Theorem 2 (Balcan *et al.* [36]) *If dis is a strongly (ϵ, γ) - good similarity function, then $\frac{4}{\gamma^2}\ln(\frac{2}{\delta})$ positive S^+ examples, and S^- negative examples are sufficient, so with probability*

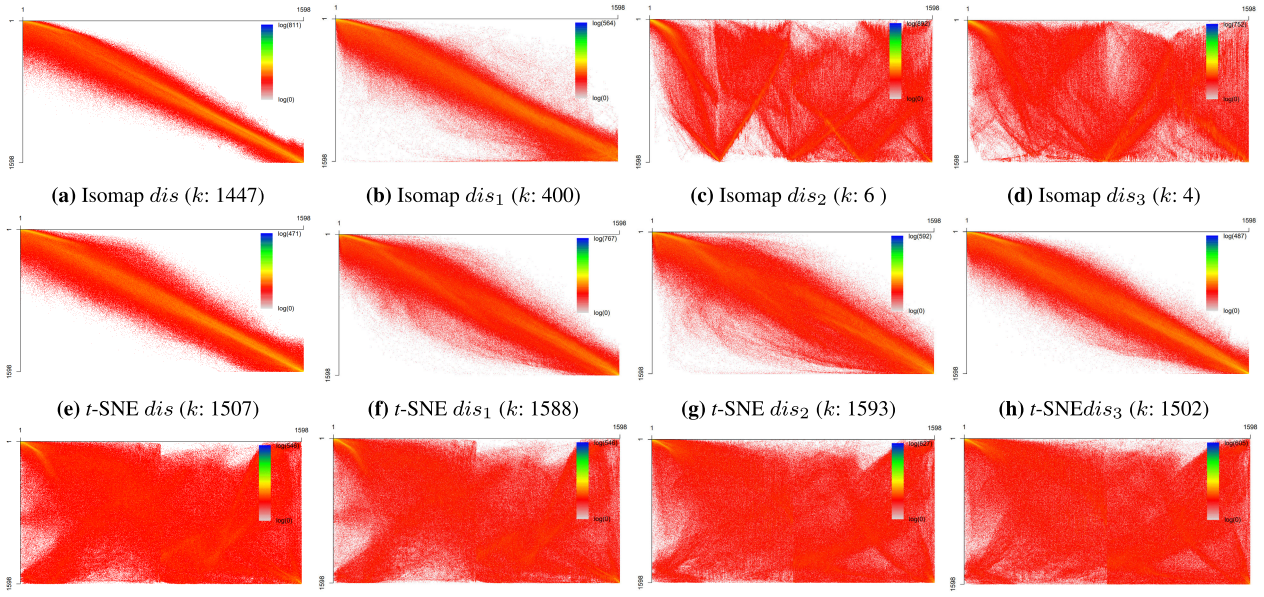


(a) Isomap dis (k: 1447) **(b)** Isomap dis_1 (k: 400) **(c)** Isomap dis_2 (k: 6) **(d)** Isomap dis_3 (k:4)

(e) t-SNE dis (k: 1507) **(f)** t-SNE dis_1 (k: 1588) **(g)** t-SNE dis_2 (k: 1593) **(h)** t-SNE dis_3 (k:1502)

(i) LE dis (k: 1000) **(j)** LE dis_1 (k: 952) **(k)** LE dis_2 (k: 795) **(l)** LE dis_3 (k: 415)

FIGURE 9. **Visualization of two-dimensional swiss roll data generated by ISOMAP, *t*-SNE and LE using as metric euclidean distance, DIS_1, DIS_2, and DIS_3.**



(a) Isomap dis (k: 1447) **(b)** Isomap dis_1 (k: 400) **(c)** Isomap dis_2 (k: 6) **(d)** Isomap dis_3 (k: 4)

(e) t-SNE dis (k: 1507) **(f)** t-SNE dis_1 (k: 1588) **(g)** t-SNE dis_2 (k: 1593) **(h)** t-SNEdis_3 (k: 1502)

(i) LE dis (k: 1000) **(j)** LE dis_1 (k: 952) **(k)** LE dis_2 (k: 795) **(l)** LE dis_3 (k: 415)

FIGURE 10. **Co-ranking matrixes of two-dimensional swiss roll data generated by ISOMAP, *t*-SNE, and LE using similarity measures euclidean distance, DIS_1, DIS_2, and DIS_3.**

$p \geq 1 - \delta$, the above algorithm produces a classifier with a maximum error of $\epsilon + \frac{\delta}{2}$.

In the work of Balcan *et al.* [36], a learning problem was specified by a labelled example (x, y) drawn from a distribution of P over $X \times \{-1, 1\}$, where X is an abstract space. In this study, the learning problem is defined by providing the low dimensional space data (y, l), (y_1, l), (y_2, l), and (y_3, l) generated by a manifold learning method M over data $X \times \{-1, 1\}$ using the dis, dis_1, dis_2, and dis_3, respectively. The objective of a learning algorithm is to produce a classification

function $g_i : Y_i \rightarrow \{-1, 1\}, i = 0 : 3$ to produce a low classification error.

In this study, we seek to discover the goodness of a similarity function in a particular learning problem. In other words, we use the same similarity function K, but in different data distribution (the low dimensional data Y, Y_1, Y_2, and Y_3 generated by the manifold learning M employing dis, dis_1, dis_2, and dis_3) having the same label l. Note that for a given i, $l(x_i) = l(y_i) = l(y_{1i}) = l(y_{2i}) = l(y_{3i})$. Consider that K is the radial basis function (RBF) kernel with formula, $K(x, x') = exp(-\frac{dis(x,x')^2}{2\sigma^2})$, *Theorem 1* states that a kernel function is a good similarity function; as such the theorems and definitions applied for similarity functions can also be applied for kernel functions. Standard algorithms such as Support Vector Machine (SVM) and Perceptron have used kernel functions to learn linear separations via computing dot products on pairs of examples. The main idea of applying kernel function is to map nonlinear data in a very high dimensional space to find a hyperplane to separate data. This study employed the RBF kernel, which is usefully used in an SVM-based classifier.

The neighbourhood structure of Y, Y_1, Y_2, and Y_3 is the same as the neighbourhood structure of X using dis, dis_1, dis_2, and dis_3, since the manifold learning M has assumed to perfectly maintain the neighbourhood structure. Thus, the K function that is assumed to be applied to the low dimensional space Y, Y_1, Y_2, and Y_3 using the squared Euclidean distance dis, can be equally applied to the high dimensional data X, but using dis, dis_1, dis_2, and dis_3, respectively. As a result, four RBF kernel functions $K(y, y')$, $K(y_1, y'_1)$, $K(y_2, y'_2)$, and $K(y_3, y'_3)$ can be reformulated as follows:

1) $K(y, y') = exp(-\frac{dis(x,x')^2}{2\sigma^2})$
2) $K(y_1, y'_1) = exp(-\frac{dis_1(x,x')^2}{2\sigma^2})$
3) $K(y_2, y'_2) = exp(-\frac{dis_2(x,x')^2}{2\sigma^2})$
4) $K(y_3, y'_3) = exp(-\frac{dis_3(x,x')^2}{2\sigma^2})$

We aim to prove that the RBF kernel can produce a lower classification error using low dimensional data Y_1, Y_2, and Y_3 than using low dimensional data Y. Let U represents the set of y that satisfy $E_{y'\sim Y}[K(y, y')|l(y) = l(y')] \geq E_{y'\sim Y}[K(y, y')|l(y) \neq l(y')] + \gamma$, and $P(U) = 1 - \epsilon$.

Proposition 3 RBF kernel K achieves a lower classification error using the low dimensional data Y_1 than using the low dimensional data Y.

Proof:

Let U_1 denotes the set of y_1 that satisfy: $E_{y'_1\sim Y_1}[K(y_1, y'_1)|l(y_1) = l(y'_1)] \geq E_{y'_1\sim Y_1}[K(y_1, y'_1)|l(y_1) \neq l(y'_1)] + \gamma$.

Since $K(y_1, y'_1) = exp(-\frac{dis_1(x,x')^2}{2\sigma^2})$, then

$E_{x'\sim X} e^{-\frac{1-e^{\frac{-dis(x,x')}{\beta}}}{2\sigma^2}} |l(y) = l(y') \geq E_{x'\sim X} e^{-(\frac{e^{\frac{dis(x,x')^2}{\beta}}}{2\sigma^2} - \alpha)} |l(y) \neq l(y') + \gamma$.

For $\alpha \geq 0.5$, we obtain $e^{\frac{dis(x,x')^2}{\beta}} - \alpha \geq 1$, and $1 - e^{-\frac{dis(x,x')^2}{\beta}} \in [0, 1[$, as such $E_{x'\sim X}[e^{-\frac{1-e^{\frac{-dis(x,x')}{\beta}}}{2\sigma^2}} |l(y) = l(y')] \geq E_{x'\sim X}[e^{-\frac{dis(x,x')^2}{2\sigma^2}} |l(y)] \neq l(y') \geq E_{x'\sim X}[e^{-\frac{e^{\frac{-dis(x,x')^2}{\beta}}}{2\sigma^2} - \alpha} |l(y) \neq l(y')] + \gamma$.

Finally, $U_1 = U \cup R_1$, where R_1 contains data samples x that satisfy $E_{x'\sim X} e^{-\frac{1-e^{\frac{-dis(x,x')}{\beta}}}{2\sigma^2}} |l(y) = l(y') \geq E_{x'\sim X} e^{-\frac{dis(x,x')^2}{2\sigma^2}} |l(y) = l(y')$, and data samples x that satisfy $E_{x'\sim X} e^{-\frac{dis(x,x')^2}{2\sigma^2}} |l(y) \neq l(y') \geq E_{x'\sim X} e^{-\frac{e^{\frac{-dis(x,x')^2}{\beta}}}{2\sigma^2} - \alpha} |l(y) \neq l(y') + \gamma$.

Therefore, $P(U_1) = P(U \cup R_1) = P(U) + P(R_1)$ as $U \cap R_1 = \varnothing$. Let's define $P(R_1) = \rho_1$, as such $P(U_1) = 1 - \epsilon + \rho_1$.

Based on *Definition 4*, RBF kernel in the low dimensional data Y_1 is a strongly $(\epsilon - \rho_1, \gamma)$-good similarity function, whereas RBF kernel in the low dimensional data Y is strongly (ϵ, γ)-good similarity function. Under the conditions of *Theorem 2*, the classification error of RBF kernel using the low dimensional data Y_1 is $\epsilon - \rho_1 + \frac{\delta}{2}$ which is lower than $\epsilon + \frac{\delta}{2}$ produced by RBF kernel low dimensional data Y.□

Proposition 4 RBF kernel K achieves a lower classification error using the low dimensional data Y_2 than using the low dimensional data Y.

Proof:

Let's define U_2 as the set of y_2 that satisfy: $E_{y'_2\sim Y_2}[K(y_2, y'_2)|l(y_2) = l(y'_2)] \geq E_{y'_2\sim Y_2}[K(y_2, y'_2)|l(y_2) \neq l(y'_2)] + \gamma$.

Since $K(y_2, y'_2) = exp(-\frac{dis_2(x,x')^2}{2\sigma^2})$, we have

$E_{x'\sim X} e^{-\frac{dis(x,x')^2}{\psi^2 2\sigma^2}} |l(y) = l(y') \geq E_{x'\sim X} e^{-\frac{dis(x,x')^2}{2\sigma^2}} |l(y) \neq l(y') + \gamma$.

$e^{-\frac{(dis(x,x'))^2}{2\psi^2\sigma^2}} = (e^{-\frac{dis(x,x')^2}{2\sigma^2}})^{1/\psi^2}, (e^{-\frac{dis(x,x')^2}{2\sigma^2}})^{1/\psi^2} \geq e^{-\frac{dis(x,x')^2}{2\sigma^2}}, \psi \geq 1$.

As a result $E_{x`\sim X}[(e^{-\frac{dis(x,x')^2}{2\sigma^2}})^{1/\psi^2} |l(y) = l(y')] \geq E_{x`\sim X}[e^{-\frac{dis(x,x')^2}{2\sigma^2}} |l(y) = l(y')] \geq E_{x`\sim X}[\frac{e^{-\frac{dis(x,x')^2}{2\sigma^2}}}{c} |l(y) \neq l(y')] + \gamma$.

On the other hand, $U_2 = U \cup R_2$, where R_2 contains data samples x that satisy $E_{x`\sim X}[(e^{-\frac{dis(x,x')^2}{2\sigma^2}})^{1/\psi^2} |l(y) = l(y')] \geq E_{x`\sim X}[e^{-\frac{dis(x,x')^2}{2\sigma^2}} |l(y) = l(y')]$.

Thus, we obtain $P(U_2) = P(U \cup R_2) = P(U) + P(R_2))$ as $U \cap R_2 = \varnothing$. Let's define $P(R_2) = \rho_2$, such that $P(U_2) = 1 - \epsilon + \rho_2$.

Based on *Definition 4*, we can say that RBF kernel in the low dimensional data Y_2 is a strongly $(\epsilon - \rho_2, \gamma)$-good similarity function, and RBF kernel in the low dimensional data Y is a strongly (ϵ, γ)-good similarity function. Under the conditions of *Theorem 2*, the classification error of RBF kernel using the low dimensional data Y_2 is $\epsilon - \rho_2 + \frac{\delta}{2}$, which is lower than $\epsilon + \frac{\delta}{2}$ produced by RBF kernel using Y. □

Proposition 5 RBF kernel K achieves a lower classification error using the low dimensional data Y_3 than using the low dimensional data Y.

Proof:

Suppose that U_3 is the set of all y_3 that satisfies: $E_{y'_3\sim Y_3}[K(y_3, y'_3)|l(y_3) = l(y'_3)] \geq E_{y'_3\sim Y_3}[K(y_3, y'_3)|l(y_3) \neq l(y'_3)] + \gamma$.

Because $K(y_3, y'_3) = exp(-\frac{dis_3(x,x')^2}{2\sigma^2})$, then

$E_{x'\sim X} e^{-\frac{dis(x,x')^2}{2\sigma^2}} |l(y) = l(y') \geq E_{x'\sim X} e^{-\frac{(dis(x,x')+dis(x,x')\mu)^2}{2\sigma^2}}$ $|l(y) \neq l(y') + \gamma$.

On the other hand, $e^{-\frac{(dis(x,x')+maxdis(x,x')\mu)^2}{2\sigma^2}} = e^{-\frac{dis(x,x')^2}{2\sigma^2}}$ $e^{-\frac{(maxdis(x,x')\mu)^2}{2\sigma^2}}$.

Let be $c = e^{\frac{(maxdis(x,x')\mu)^2}{2\sigma^2}}$, and $c \geq 1$, then $E_{x`\sim X}[e^{-\frac{dis(x,x')^2}{2\sigma^2}} |l(y) = l(y')] \geq E_{x`\sim X}[\frac{e^{-\frac{dis(x,x')^2}{2\sigma^2}}}{c} |l(y) \neq l(y')] + \gamma$.

On the other hand, $U_3 = U \cup R_3$, where R_3 contains the x data samples that satisy $E_{x'\sim X}[\frac{e^{-dis(x,x')^2}}{c} |l(y) \neq l(y')] + \gamma \leq E_{x'\sim X}[e^{-\frac{dis(x,x')^2}{2\sigma^2}} |l(y) = l(y')] \leq E_{x'\sim X}[e^{-\frac{dis(x,x')^2}{2\sigma^2}} |l(y) \neq l(y')] + \gamma$.

As a result, $P(U_3) = P(U \cup R_3) = P(U) + P(R_3))$ as $U \cap R_3 = \varnothing$. We define $P(R_3) = \rho_3$, such that $P(U_3) = 1 - \epsilon + \rho_3$.

Based on *Definition 4*, we have proved that RBF kernel in the low dimensional data Y_3 is strongly $(\epsilon - \rho_3, \gamma)$-good similarity function, whereas RBF kernel applied in the low dimensional data Y is strongly (ϵ, γ)-good similarity function. Under the conditions of *Theorem 2*, the classification error of RBF kernel using the low dimensional data Y_3 is $\epsilon - \rho_1 + \frac{\delta}{2}$, which is lower than $\epsilon + \frac{\delta}{2}$ produced by RBF kernel using the low dimensional data Y.□

Overall, RBF kernel applied in the low dimensional data generated by a manifold learning M[8] using dissimilarity measures dis_1, dis_2, and dis_3 can help a learning problem to achieving lower classification errors than RBF kernel applied in the low dimensional data generated by the manifold learning M using the Euclidean distance dis.

VI. CONCLUSION AND FURTHER WORKS

Supervised manifold learning has been used in many scenarios to achieve higher classification accuracy and provide better visualization. This paper provides a theoretical analysis of the impact of dissimilarity measure on manifold learning regarding classification error. Dissimilarity measure forces relocating data samples using class information, but it does not improve data structure capturing. Following the theoretical analysis and supported by experimental results, we can conclude that the dissimilarity measure in Isomap, t-SNE and LE worsens data structure capturing. Therefore, it would be more useful to use Euclidean distance than dissimilarity measures. However, dissimilarity measure dis_3 has a positive impact on t-SNE, which can help preserve global data information better. In addition, a dissimilarity measure can be usefully incorporated in manifold learning techniques to achieve a better RBF-based classifier, and the class-separation achieved by supervised dimensionality reduction methods can reduce the classification error.

Overall, supervised manifold learning can be used for classification purposes with the advantage of classification error reduction. In visualization, the class information involved in dissimilarity measure can destroy data structure capturing. As a result, incorrect information can be obtained from two-, three-dimensional visualizations, which can lead us to make a wrong decision. However, we strongly advise against using supervised manifold learning/dimensionality reduction techniques as a pre-processing step of classification. Still, we strongly advise not using supervised manifold learning for visualization purposes as the two-dimensional representation using supervised manifold learning does not improve the preservation of neighbourhood structure, but instead, destroys it.

Proving that a dissimilarity function could help any classification method (kernel-based or not) to achieve a lower classification error is an objective for further work.

[8]Note that manifold learning M perfectly preserves the neighborhood structure using dis, dis_1, dis_2, and dis_3.

REFERENCES

[1] H. Hotelling, "Analysis of a complex of statistical variables into principal components," *J. Educ. Psychol.*, vol. 24, no. 6, p. 417, 1933.

[2] J. B. Kruskal, "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis," *Psychometrika*, vol. 29, no. 1, pp. 1–27, Mar. 1964.

[3] Y. Ma and Y. Fu, *Manifold Learning Theory and Applications*. Boca Raton, FL, USA: CRC Press, 2011.

[4] J. B. Tenenbaum, V. De Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[5] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, Dec. 2000.

[6] M. M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2002, pp. 585–591.

[7] D. Donoho and C. Grimes, "Hessian eigenmaps: New tools for nonlinear dimensionality reduction," *Proc. Nat. Acad. Sci. USA*, vol. 100, pp. 5591–5596, Mar. 2003.

[8] X. Xing, S. Du, and K. Wang, "Robust hessian locally linear embedding techniques for high-dimensional data," *Algorithms*, vol. 9, no. 2, p. 36, May 2016.

[9] Z. Zhang and H. Zha, "Principal manifolds and nonlinear dimensionality reduction via tangent space alignment," *SIAM J. Sci. Comput.*, vol. 26, no. 1, pp. 313–338, Jan. 2004.

[10] K. Q. Weinberger and L. K. Saul, "Unsupervised learning of image manifolds by semidefinite programming," *Int. J. Comput. Vis.*, vol. 70, no. 1, pp. 77–90, Oct. 2006.

[11] S. Lafon and A. B. Lee, "Diffusion maps and coarse-graining: A unified framework for dimensionality reduction, graph partitioning, and data set parameterization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 9, pp. 1393–1403, Sep. 2006.

[12] S. Lafon, Y. Keller, and R. R. Coifman, "Data fusion and multicue data matching by diffusion maps," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 11, pp. 1784–1797, Nov. 2006.

[13] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.

[14] G. Rosman, M. M. Bronstein, A. M. Bronstein, and R. Kimmel, "Nonlinear dimensionality reduction by topologically constrained isometric embedding," *Int. J. Comput. Vis.*, vol. 89, no. 1, pp. 56–68, Aug. 2010.

[15] J. Chen, Z. Ma, and Y. Liu, "Local coordinates alignment with global preservation for dimensionality reduction," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 1, pp. 106–117, Jan. 2013.

[16] L. McInnes, J. Healy, and J. Melville, "UMAP: Uniform manifold approximation and projection for dimension reduction," 2018, *arXiv:1802.03426*. [Online]. Available: https://arxiv.org/abs/1802.03426
[17] L. Hajderanj, D. Chen, E. Grisan, and S. Dudley, "Single-and multi-distribution dimensionality reduction approaches for a better data structure capturing," *IEEE Access*, vol. 8, pp. 207141–207155, 2020.
[18] S. Kadoury, "Manifold learning in medical imaging," in *Manifolds II-Theory and Applications*. London, U.K.: IntechOpen, 2018.
[19] K. Seo, R. Pan, D. Lee, P. Thiyyagura, and K. Chen, "Visualizing Alzheimer's disease progression in low dimensional manifolds," *Heliyon*, vol. 5, no. 8, Aug. 2019, Art. no. e02216.
[20] Y. Huang, G. Kou, and Y. Peng, "Nonlinear manifold learning for early warnings in financial markets," *Eur. J. Oper. Res.*, vol. 258, no. 2, pp. 692–702, Apr. 2017.
[21] X. Geng, D.-C. Zhan, and Z.-H. Zhou, "Supervised nonlinear dimensionality reduction for visualization and classification," *IEEE Trans. Syst., Man, Cybern., B, Cybern.*, vol. 35, no. 6, pp. 1098–1107, Dec. 2005.
[22] L. Hajderanj, I. Weheliye, and D. Chen, "A new supervised t-SNE with dissimilarity measure for effective data visualization and classification," in *Proc. 8th Int. Conf. Softw. Inf. Eng.*, Apr. 2019, pp. 232–236.
[23] M. Vlachos, C. Domeniconi, D. Gunopulos, G. Kollios, and N. Koudas, "Non-linear dimensionality reduction techniques for classification and visualization," in *Proc. 8th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, Jul. 2002, pp. 645–651.
[24] C. Wei, J. Chen, and Z. Song, "Developments of two supervised maximum variance unfolding algorithms for process classification," *Chemometric Intell. Lab. Syst.*, vol. 159, pp. 31–44, Dec. 2016.
[25] D. De Ridder and R. P. Duin, "Locally linear embedding for classification," Dept. Imag. Sci. Technol., Pattern Recognit. Group, Delft Univ. Technol., Delft, The Netherlands, Tech. Rep. PH-2002-01, 2002, pp. 1–12.
[26] A. Cheriyadat and L. M. Bruce, "Why principal component analysis is not an appropriate feature extraction method for hyperspectral data," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, vol. 6, Jul. 2003, pp. 3420–3422.
[27] Y. LeCun and C. Cortes. (1998). MNIST Handwritten Digit Database. AT&T Labs. [Online]. Available: http://yann.lecun.com/exdb/mnist
[28] *SEER, Surveillance, Epidemiology, and End Results Program: Overview of the SEER Program (1973-2014)*. Accessed: Aug. 2, 2017. [Online]. Available: https://seer.cancer.gov/about/overview.html
[29] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers, "ChestXray8: Hospital-scale chest X-ray database and benchmarks on weakly supervised classification and localization of commonthorax diseases," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3462–3471.
[30] S.-Q. Zhang, "Enhanced supervised locally linear embedding," *Pattern Recognit. Lett.*, vol. 30, no. 13, pp. 1208–1218, Oct. 2009.
[31] M. Yu, S. Zhang, L. Zhao, and G. Kuang, "Deepsupervised t-SNE for SAR target recognition," in *Proc. 2nd Int. Conf. Frontiers Sensors Technol. (ICFST)*, Apr. 2017, pp. 265–269.
[32] J. Cheng, H. Liu, F. Wang, H. Li, and C. Zhu, "Silhouette analysis for human action recognition based on supervised temporal t-SNE and incremental learning," *IEEE Trans. Image Process.*, vol. 24, no. 10, pp. 3203–3217, Oct. 2015.
[33] T. Ghosh and M. Kirby, "Supervised dimensionality reduction and visualization using centroid-encoder," 2020, *arXiv:2002.11934*. [Online]. Available: https://arxiv.org/abs/2002.11934
[34] A. J. Isenmann, *Modern Multivariate Statistical Techniques*. New York, NY, USA: Springer, 2008.
[35] J. A. Lee and M. Verleysen, "Quality assessment of dimensionality reduction: Rank-based criteria," *Neurocomputing*, vol. 72, nos. 7–9, pp. 1431–1443, Mar. 2009.
[36] M.-F. Balcan, A. Blum, and N. Srebro, "A theory of learning with similarity functions," *Mach. Learn.*, vol. 72, nos. 1–2, pp. 89–112, Aug. 2008.

LAURETA HAJDERANJ received the M.Sc. degree in mathematics and informatics engineering from the University of Tirana, Tirane, Albania, in 2012. In 2018, she received the M.Sc. degree in internet and database systems from London South Bank University. She is currently pursuing the Ph.D. degree in computer science with London South Bank University. From 2013 to 2015, she worked as a Lecturer with the Computer Science Department, University of Aleksander Moisiu, Durres, Albania. Her research interests include high dimensional data embedding and visualization, manifold learning, structure capturing, and acceleration algorithms.

DAQING CHEN (Member, IEEE) received the bachelor's degree in systems engineering from Northwestern Polytechnical University, Xian, China, in 1982, and the M.Phil. degree in automatic control engineering from the National University of Defense Technology, Changsha, China, in 1990, and the Ph.D. degree in automatic control engineering from Northwestern Polytechnical University, in 1993. From 1994 to 1997, he worked as a Postdoctoral Researcher and then an Associate Professor with the National Key Laboratory of Radar Signal Processing, Xidian University, Xian. From 1997 to 1998, he was a Research Associate with the Department of Computer Science and Engineering, The Chinese University of Hong Kong. From 1998 to 1999, he worked as a Research Fellow with the System, Electronics, and Information Laboratory, IRESTE, University of Nantes, Nantes, France. Since 1999, he has been working with London South Bank University. He is currently a Senior Lecturer in informatics with the School of Engineering. His research interests include deep learning algorithms with applications in lip reading, medical image diagnosis, high dimensional data embedding and visualization, high-volume data labeling, and business intelligence.

ISAKH WEHELIYE received the bachelor's degree in software engineering from Kingston University, London, U.K., in 2007, the M.Sc. degree in information technology management from London South Bank University, U.K., in 2015. He is currently pursuing the Ph.D. degree in computer science with London South Bank University, where he works as an Assistant Tutor. From 2008 to 2015, he worked as a Software Developer. His research interests include deep learning architectures and algorithms with application in medical image diagnosis, data analysis, data visualization, and multi-instance-multi-label classification algorithms.