Journal of Theoretical and Applied Information Technology
30th April 2023. Vol.101. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A SYSTEMATIC REVIEW OF VARIOUS NONLINEAR DIMENSIONALITY REDUCTION TECHNIQUES FOR ANALYSIS OF BIG DATA

OLAIYA FOLORUNSHO[1,2], TINY DU TOIT[1]

[1]Unit for Data Science and Computing North-West University, Potchefstroom Campus, South Africa
[2]Department of Computer Science Federal University Oye Ekiti Oye Ekiti, Ekiti State, Nigeria
E-mail: 44698062@nwu.ac.za, Tiny.DuToit@nwu.ac.za

ABSTRACT

The recent advancement in data-driven computing technologies in various disciplines has resulted in massive dimensional data being collected from multiple information sources. Many machine learning problems require processing a large set of features, finding it challenging to analyse the training set and find a suitable solution that maximises predictive power for the classifier performance. Therefore, it has become imperative to reduce massive features to the most significant ones for accurate data analysis from computation devices for predictive purposes. Over the years, researchers have developed several linear dimension reduction methods to reduce data dimensionality and identify data points with the highest possible variance. However, such techniques could not effectively handle data with a nonlinear relationship among the variables. Therefore, this paper presents state-of-the-art nonlinear dimensionality reduction methods for modelling complex nonlinear structures. The paper is presented in four folds: The first step involves discussing the most used nonlinear dimensionality reduction techniques. Second, a summary of the scope of application areas where dimensionality reduction methods have been applied is presented. The third fold compares various techniques based on their challenges and advantages. Finally, the performance evaluation of each approach in terms of its suitability for various applications will be discussed. The paper concluded that the autoencoder is an excellent technique for the dimensionality reduction of nonlinear high-dimensional data based on its tendency to accurately reconstruct data if there is a nonlinear connection in the feature space, also acurately capture the manifold's topology, and it tends to capture more of the global properties than other global techniques.

Keywords: *Big Data, Data Analysis, Dimensionality Reduction, Features, Nonlinear Techniques. Techniques.*

1. INTRODUCTION

Over the recent decades, due to technological advancement and digital transformation, the volume of data generated from different application domains such as education, the world wide web, social media, business, medicine and so on has continued to increase in size, complexity, and dimensionality [1]. Consider a dataset represented either as a database table or matrix, with each row being a collection of attributes that describe a specific instance of something. When there are many attributes, the space of distinct possible rows increases exponentially. Thus, sampling the space gets more challenging the higher the dimensionality. In addition, the high time complexity of algorithms processing high-dimensional data may lead to many issues.

In most cases, machine learning (ML) algorithms struggle with massive amounts of data. This shortcoming is known as the curse of dimensionality. Generally, the performance of ML algorithms increases as the number of input characteristics reduces due to the elimination of redundant features [2],[3],[4]. ML algorithms can produce more accurate predictions and efficient data analysis with fewer dimensions. Reducing its dimensionality is necessary to effectively model the massive dimensional data in real-world applications [5],[6]. Dimensionality reduction (DR) serves as the most popular approach to circumvent the curse of dimensionality [7],[8].

Data is transformed from a high dimensional space to a low dimensional space through DR while retaining most of its essential properties [8]. The dimensionality curse is usually avoided by representing the original data in a low-dimensional form that is easy to analyse, process, and visualise [9]. The DR is a step in the preprocessing phase of data mining and knowledge discovery, which helps remove noisy, irrelevant data and redundant features from the dataset, decreasing computational time, improving algorithm efficiency, and identifying data points with the highest possible variance [10],[11]. Implementing DR via feature extraction and selection reduces the time complexity and computational resources and improves ML algorithms' overall performance [12].

Depending on the data used, the DR approach can be categorised into linear and nonlinear DR. In past decades, several linear dimensionality reduction techniques (LDRTs) such as Linear Discriminant Analysis (LDA) [13], Truncated Singular Value Decomposition (TSVD) [14], Factor Analysis (FA) [15], and Principal Component Analysis (PCA) [16],[17] have been used for DR. All these methods could not effectively handle data that possess nonlinear relationships. However, many nonlinear dimensionality reduction techniques (NLDRTs) have recently been introduced to handle complex nonlinear data. The techniques include Kernel PCA [18],[19], Multidimensional Scaling (MDS) [20], t-distributed Stochastic Neighbour Embedding (t-SNE) [21], Local Linear Embedding (LLE) [22],[23], Isometric mapping (Isomap) [24] and the autoencoder [25],[26]. The NLDRTs may offer a comparative advantage over the LDRTs because of the tendency to characterise the nonlinear manifold relationship between data points [27]. Previous studies revealed that NLDRTs resulted in better dimension reduction, accuracy, and performance than the linear methods while experimenting with complex machine learning tasks [28].

The remainder of the paper is organized according to the following outline. The description of an overview of the field of application of DRTs is presented in Section 2. A taxonomy of NLDRTs is presented in Section 3 while Section 4 addresses the challenges of NLDRTs. Lastly, Section 5 presents the conclusions.

2. AN OVERVIEW OF THE FIELDS OF APPLICATION OF DIMENSIONALITY REDUCTION TECHNIQUES

Different DRTs and their variations have been developed to address numerous issues in the computational domains. Each technique solves different problems and has its pros and cons [29]. This section presents some areas of science or technology where high-dimensional data are frequently encountered.

2.1. Image Processing

The steps involved in processing an image are acquisition, pre-processing, segmentation, representation, and description, followed by interpretation and recognition [30]. Digital images can be divided into four categories: binary, greyscale, indexed, and true colour (RGB) [31]. Text, fingerprints, and architectural plans are examples of images with binary representations, where each pixel is either black or white. Greyscale images commonly range in the shade from 0 to 255, including X-rays, pictures of printed objects, and other images [32]. Furthermore, there are indexed graphics that have a colour map attached that lists every colour that was used to create the image. Finally, true colour, or RGB images, is used to describe each pixel's red, green, and blue composition. The amount of digital information has increased significantly over time. However, traditional computer systems often ignore research in image databases. This trend has spawned due to the enormous quantity of data needed to represent images and the difficulty of automatically analysing images [33].

2.2. Data Mining

Data mining (DM) is the process that automatically generates and extracts implicit and prospective patterns from large datasets [34]. DM can be applied in various areas, e.g. business, agriculture, science, and engineering. Much data is generated, and there is a need to derive knowledge patterns from large information-rich datasets. The

data mining task has five primary foci: description, prediction, classification, association, and clustering [35].

The current growth of big data, in terms of the size of records and features, has created significant issues for DM and general data science, despite recent technological breakthroughs in data processing and computer science [36]. Big data generates multidimensional datasets because the size of the data seems to be so massive. An extensive data set with several dimensions make it difficult to analyse or search for patterns in the data. Depending on the process one is interested in, high-dimensional data can be obtained from various sources.

2.3. Process of Sensor Arrays

The sensor array process uses multiple identical sensors in various applications [37]. They typically gather and process electromagnetic signals in a specific geometric pattern. Acoustic arrays, antenna arrays, and traditional beamformers are a few examples of sensor arrays. One benefit of utilizing a sensor array instead of a single sensor is that it provides additional dimensions to the observation, making it easier to analyse more factors and increasing the precision of predictions. An array of radio antenna components used for beamforming, for example, can raise antenna gain in the signal's direction while decreasing the gain in other directions to improve the signal-to-noise ratio by coherently amplifying the signal. DRTs reduce the computational complexity of the direction-of-arrival estimate from the sensor arrays [38]. DR approaches enable linear transformations to convert full-dimension data into a lower-dimensional space while reducing the required computations.

2.4. Multivariable Data Analysis

Multivariate data analysis is a statistical analysis technique that uses more than two dependent variables and produces a single result. Since everything in the world occurs for various reasons, many situations in daily life can serve as real-world instances of multivariate equations [39]. This explains why multivariate difficulties are prevalent in the actual world. For example, depending on the season, one can anticipate the weather for any given year. Multivariate statistical methods fall into two categories: dependency methods, which look at cause-and-effect relationships between variables, and interdependence methods, which examine the cause-and-effect interactions between variables [40]. Data with multiple variables have moderate to high dimensions, and analysis can be challenging (e.g., when using statistical methods); reducing the dataset might assist by making it easier to analyse [41]. Therefore, DR finds fewer variables or removes the least essential variables from the multivariable data, removing some noise, reducing the model's complexity, and helping to mitigate overfitting on the data.

3. TAXONOMY OF NONLINEAR DIMENSIONALITY REDUCTION TECHNIQUES

This section introduces various classifications of NLDRTs with their respective subclasses and examples of popular algorithms. The NLDRTs transform and use the most relevant feature combinations, reducing space and times demands. Fig. 1 shows a taxonomy subdividing the NLDRTs as follows: (1) Preserving global properties techniques seek to retain the global properties of the given data from its original higher dimensional space into the lower dimensional space, (2) global alignment of the nonlinear models which compute several local nonlinear models and align the nonlinear models globally, and (3) preserving local properties attempt to keep local properties of the higher dimensional space into the low-dimensional representation.

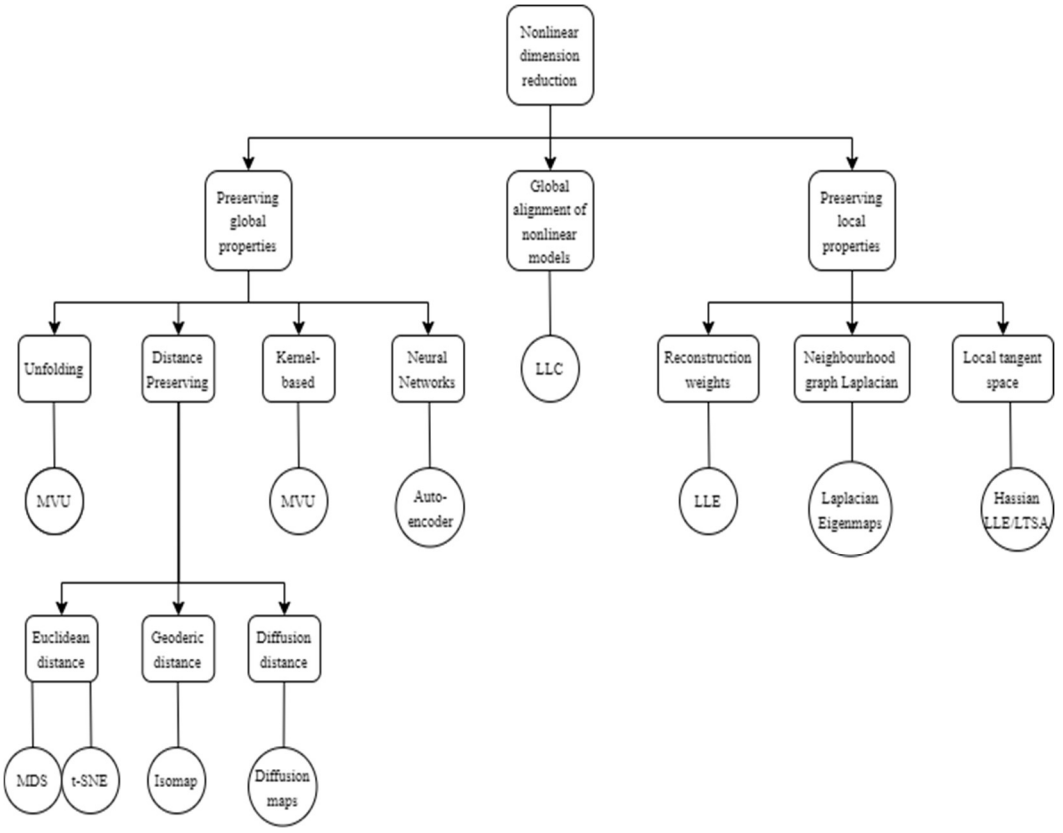

Fig. 1. Taxonomy Of Nonlinear Dimensionality Techniques

3.1. Preserving Global Properties

This type of NLDRT attempts to retain the global properties of the data set. The four categories of preserving global properties are (1) Unfolding, (2) Distance preservation, (3) Kernel-based, and (4) Neural networks. This subsection discusses all six techniques under the respective subclasses of preserving global properties techniques.

3.1.1. Maximum Variance Unfolding

Maximum Variance Unfolding (MVU), previously referred to as Semidefinite Embedding, is a semidefinite programming method that reduces the large dimensionality of vectorial input nonlinearly by learning the kernel matrix [42]. By creating a neighbourhood on the data (similar to Isomap) and keeping pairwise distances in the resulting graph, MVU learns the kernel matrix. In contrast to the Isomap approach, which retains geodesic distances, the MVU method learns the data from similarities. MVU preserves local distances and angles between the pairs of all neighbours of each data point in the data set. To increase the Euclidean distance between the data points (i.e., with the condition that the local geometric of the data manifold is not deformed), MVU restricts the distance in the neighbourhood graph to remain unchanged [27]. By maximizing the variance of the embeddings while keeping the original data's local distances, MVU effectively reduces the data's dimensionality and produces a low-dimensional representation. Each data point x_i, and its k nearest neighbours X_{ij} $(j = 1, 2, \dots, k)$ are initially connected by MVU to construct a neighbourhood graph, G. With the constraint that the distances within the neighbourhood graph G be retained, MVU aimed to maximise the sum of the squared Euclidean distance between the various data points.

3.1.2. Multidimensional Scaling

Multidimensional scaling (MDS) is the NLDRT used when there is a difference between the paired distances in the original space and the pairwise distances in the low-dimensional space [43]. When lowering nonlinear data's dimensionality, MDA ensures that the distances between instances are retained. MDA attempts to decrease the dimensionality of nonlinear data while maintaining the distances between instances. Metric and non-metric MDS algorithms are available. The distinctions between both originate from the kinds of data they are intended to deal with, even though both seek to discover the optimum lower-dimensional representation of high-dimensional data.

Suppose the pairwise distance between two points is known. In that case, MDS will retain it by projecting the points into a low-dimension space, preserving the pairwise distances there as near as possible to the pairwise distances in the original space. Using data on interpoint distances creates a configuration of points in Euclidean space [44]. MDS comes in two forms, and they share the same fundamental ideas. The metrics and calculations are utilized to account for the difference. The MDS input is the distance matrix representing the separations between object pairs. The distance between data points in the reduced dimension d_{ij}, which is nearly equivalent to the real distance according to the distance matrix d_{ij}, is represented by the distance matrix in the needed dimension. A linear (classical/metric) or monotonic connection can exist between actual data distances and discrepancies (non-metric). The conventional approach is the best alternative when the distance matrix correctly represents the Euclidean distance between two locations. Non-metric MDS does NLDR by using distances that can be understood in an ordinal manner. The method's efficacy is estimated based on the discrepancy between actual and anticipated distances.

3.1.3. T-distributed Stochastic Neighbour Embedding

T-distributed Stochastic Neighbour Embedding (t-SNE) is an unsupervised manifold algorithm that was introduced by [45]. It is an excellent NLDRT for visualising high-dimensional data by assigning a position to each data point in a two- or three-dimensional map [46]. The t-SNE is an improved variation of SNE that has a long-tailed distribution and is useful for embedding high-dimensional data for visualization in a two- or three-dimensional low-dimensional space [47]. The t-SNE specifically represents each high-dimensional object by a two- or three-dimensional point, intending to model comparable objects by nearby points and dissimilar objects, with a high likelihood, by distant points [48]. The t-SNE technique consists of two main steps. In the first step, t-SNE creates a probability distribution between pairs of high-dimensional objects, giving more probability to similar objects and less probability to dissimilar ones. The second step involved defining a comparable probability distribution over the points in the low-dimensional map, where t-SNE minimizes the Kullback-Leibler divergence between the two distributions about the positions of the points in the map [49]. Even though t-SNE plots frequently seem to display clusters, the parameterisation employed can have a significant impact on the visual clusters; hence, a detailed understanding of the t-SNE parameters is necessary [50].

The conditional probability $P_{j|i}$, defined as it is expressed in Equation (1), expresses how similar data points x_j to data point x_i are:

$$P_{j|i} = \frac{exp(\frac{-\|x_i - x_j\|^2}{2\sigma_1^2})}{\sum_{k \neq i} exp(\frac{-\|x_i - x_k\|^2}{2\sigma_1^2})} \quad (1)$$

The probability in the original space is expressed in Equation (2)

$$P_{j|i} = \frac{(P_{i|j} + P_{j|i})}{2n} \quad (2)$$

where n is the data set's size.

3.1.4. Isometric Mapping

Isometric mapping (Isomap) is a nonlinear technique used for DR based on the spectral principle that preserves geodesic distances in lower dimensions. Through isometric mapping, Isomap

performs NLDR. It is a refinement of Kernel PCA or MDS [44]. A neighbourhood network is established in the initial step. Then, using graph distance, it tries to determine the estimated geodesic distance between each pair of locations. The low dimensional embedding of the data set is discovered in the third stage, which decomposes the eigenvalues of the geodesic distance matrix. Isomap does not over-emphasize the space between clusters, in contrast to t-SNE. Thus, it produces more suitable distance measures between various classes to analyse different trajectories that ordinarily do not fit into a traditional cluster. In nonlinear data, geodesic distances are superior to Euclidean distances in terms of accuracy and reduction in dimensionality. Furthermore, nonlinear interactions between cells are preserved via Isomap.

3.1.5. Diffusion Map

Coifman and Lafon developed the DR algorithm known as the Diffusion map [51]. When employing a diffusion map, a data set is converted into a family of embeddings in Euclidean space. These embeddings are typically low-dimensional, and their coordinates may be calculated using the eigenvectors and eigenvalues of a diffusion operator on the data. In a diffusion map, low-dimensional data is often embedded in higher-dimensional spaces [50]. The data is located on a possible non-linear geometric structure or manifold. Diffusion distance computation is computationally intensive. It is feasible to map data points into Euclidean space using the diffusion metric. The Euclidean distance takes the place of the diffusion distance in the data space in this new diffusion space. [52]. A diffusion map reorganises data by mapping coordinates between data and diffusion space. To minimise dimensionality, one need to take advantage of it. In the new space, diffusion maps and distances preserve the inherent geometry of data sets and are measured on a lower-dimensional structure, so data points will require fewer coordinates.

3.1.6. Kernel PCA

PCA has been an excellent technique for modelling linear features in high-dimensional data. However, many high-dimensional data sets are nonlinear [53]. Therefore, PCA cannot correctly model the dispersion of the data since high dimensional data is approximately near a nonlinear manifold. Consequently, kernel PCA was designed as one of the algorithms to model NLDR. With the kernel in kernel PCA, principal space components in the high dimensional feature space could be computed efficiently with the input space of some nonlinear mapping [54].

Kernel PCA is a nonlinear normal PCA that uses kernels [55]. When dealing with nonlinear data sets, kernel PCA performs effectively where the normal PCA cannot function. In Kernel PCA, the data is processed first by a kernel function and then projected onto a new, higher-dimensional feature space where the classes remain linearly separable. The data would then be projected back into a lower-dimensional space by the algorithm using the standard PCA. In this manner, Kernel PCA converts nonlinear data into a lower-dimensional data space that can be used with linear classifiers.

Kernel PCA addresses this issue by using a nonlinear mapping function (i.e., a kernel) to transform the data into a higher-dimensional feature space, where the data may become more separable. The principal components are then computed in this feature space, rather than the original input space. The kernel function measures the similarity between pairs of data points in the input space and maps them to a new feature space. The mapping is chosen so that the inner products of the transformed data correspond to the similarity measures in the input space [56].

The kernel matrix K of the datapoints x_i is computed via Kernel PCA. The kernel matrix's entries are described by

$$k_{ij} = k(X_i,\ X_j) \qquad (3)$$

where every function that results in a positive-semidefinite kernel k is a candidate for the role of kernel function k.

3.1.7. Autoencoder

An autoencoder (AE) is an unsupervised artificial neural network (ANN) that learns how to effectively compress and encode data before reconstructing the data from the compressed, encoded version to a representation that is

comparable to the original input [57],[58]. The goal of the autoencoder is to minimize the reconstruction error between the original input and the reconstructed output. By doing so, the autoencoder learns to extract meaningful features from the input data and use them to reconstruct it [59]. AEs have become increasingly popular in recent years due to their ability to learn useful representations of data in an unsupervised manner, without requiring labelled training data [60].

A schema of the typical diagram of an AE is illustrated in Fig. 2. The AE has three essential parts: an encoder, a code and a decoder. The input passes through the encoder part, which then compresses and stores it in the code layer, and then the decoder decompresses it back to its original state. In order for an autoencoder to work properly, its output must be nearly identical to its input. The AE has been of great importance in ML because of its promptness to train networks to eliminate noise from input data and reconstruct their input data. The AE layers may not necessarily be symmetrical in the sense of weight matrices or activation functions [28]. The basic AE is the foundation for more complex architectures that perform feature extraction and classification. AEs have been recently employed as one of the nonlinear feature methods and have resulted in better DR accuracy performance than other DRTs [28].

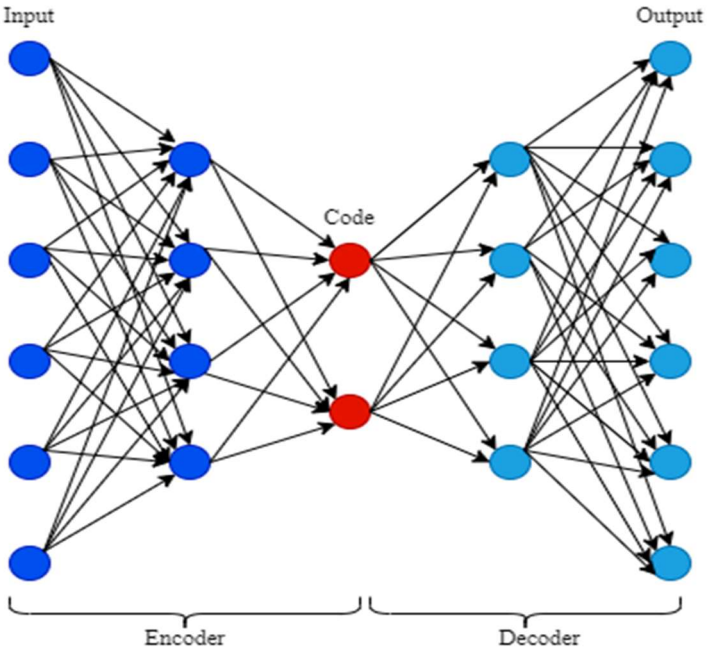

Fig. 2. Schema Of A Basic Autoencoder

The input to the autoencoder is a vector $x \in R^{d_z}$. The latent variable single hidden layer of the encoder and the reconstruction of the inputs back from the latent space is given by equations (4) and (5):

$$z = f_d(W_d x + b_d) \quad (4)$$

$$\hat{x} = f_e(W_e z + b_e) \quad (5)$$

where x, and $\hat{x} \in \mathbb{R}^n$ denote the input and output, f represents the activation function, z denotes the hidden layer, W_d and W_e represent the weights of the hidden and output layers, and the biases of the two layers are indicated by b_d and b_e respectively.

3.2. Global Alignment of Nonlinear Models

Global Alignment of nonlinear models performs global alignment of nonlinear models by

combining and computing several locally nonlinear models. A significant example of this class is LLC, which preserved the neighbourhood mapping of high dimensional observation to low dimensional vector using a global cost function with an analytical solution.

3.2.1 Local Linear Embedding

Local Linear Embedding (LLE) aims to preserve the local structure of high-dimensional data in a lower-dimensional space [61]. LLE was introduced by Sam T. Roweis and Lawrence K. Saul in a paper titled "Nonlinear Dimensionality Reduction by Locally Linear Embedding" in 2000 [62]. The method involves first defining a neighbourhood around each data point in the high-dimensional space. The neighbourhood is defined by selecting the k nearest neighbours of each point, where k is a user-defined parameter. LLE then seeks to find a lower-dimensional representation of the data that preserves the pairwise relationships between the neighbours within each neighbourhood [63].

To achieve this, LLE constructs a weight matrix W that describes the linear relationships between neighbouring points, and then seeks a low-dimensional embedding Y that minimizes the reconstruction error of the data points from their weighted linear combination in the low-dimensional space [64]. However, LLE has some limitations. It can be computationally expensive for large datasets, and its performance can be sensitive to the choice of the neighbourhood size parameter [45]. Also, LLE assumes that the data lies on a low-dimensional manifold, which may not be true for some datasets [65].

A global alignment of the nonlinear models is carried out following the computation of many locally nonlinear models by [66]. This procedure consists of two steps: first, using the Expectation Maximization (EM) method, a variety of local nonlinear models are built on the data; second, the local nonlinear models are aligned to create the low-dimensional data representation using a variation of Local Linear Coordination (LLE).

Overall, while LLE is a powerful method for preserving local structure in high-dimensional data, it is important to consider its limitations and use it appropriately. For example, LLE may be best suited for smaller datasets where the neighbourhood size can be more carefully tuned and where the data is more likely to lie on a low-dimensional manifold.

3.3. Preserving Local Properties

Preserving local properties is the class of NLDRTs that attempts to maintain the data's local properties. The three classes of preserving local properties are (1) Reconstruction weights, (2) Neighbourhood graph Laplacian, (3) and Local tangent space. This section discusses all three techniques under the respective subclasses of preserving local properties techniques.

3.3.1. Local Linear Embedding

Local Linear Embedding (LLE) is a technique for unsupervised learning that creates low-dimensional embeddings from high-dimensional inputs, connecting each training instance to its nearest neighbour [67]. LLE, Isomap, and MVU all create graph representations of the data points, which makes them comparable. Unlike Isomap, LLE does not require locally linear fits to pairwise distance estimates from nonlinear structures [68]. To identify nonlinear structures in high-dimensional data, LLE takes advantage of the local symmetries of linear reconstruction. LLE does not use local minima and converts its input to a single low-dimensional global coordinate system [67]. If the input data is in the form of d-dimensional vectors $\vec{X}_i$ the neighbours of each data point, $\vec{X}_i$ are first calculated via LLE. A locally linear area of the manifold, defined by the points and their neighbours, allows for reconstructing each data point from its neighbours. The reconstruction errors can be measured using equation (6):

$$\epsilon(W) = \sum_i(\left|\vec{Y}_i - \sum_j W_{ij}\vec{X}_j\right|)^2 \qquad (6)$$

where W_{ij} is the weight that describes the contribution of the j^{th} data point to the i^{th} reconstruction. By calculating the dimensional coordinates of $\vec{Y}_i$ that minimize the cost function, the data vector $\vec{X}_i$ is transformed into a low-dimensional space

$$\Phi(W) = \sum_i(|\vec{X}_i - \sum_j W_{ij}\vec{X}_j|)^2. \quad (7)$$

The low-dimensional vectors $\vec{Y}_i$ are best reconstructed by the weights, by minimizing the cost function in equation (7).

The local neighbourhoods of data points are used to calculate the reconstruction weights, whilst eigen decomposition is used to get the low-dimensional embedding. Finally, the eigenvectors are used to compute each dimension in the reduced dimension space iteratively.

3.3.2. Laplacian Eigenmaps

Laplacian Eigenmaps (LE) locate a low-dimensional data representation while maintaining the manifold's local attributes [69]. Pairwise distances between close neighbours determine local attributes in LE. By minimising the distances between each data point and its k nearest neighbours, LE is used to represent the data in a low-dimensional manner. This is done in a weighted way so that the distance between a data point and its first nearest neighbour in a low-dimensional data representation affects the cost function more than the distance between the data point and its second nearest neighbour [68]. The minimisation of the cost function is referred to as an eigenproblem in spectral graph theory.

Every data point x_i in the neighbourhood graph G created by the LE algorithm is connected to its k closest neighbours. The weight of each edge connecting G points x_i and x_j in graph G is calculated using the Gaussian kernel function, resulting in a sparse adjacency matrix W. The cost function minimized in calculating the low-dimensional representations is given by equation (8):

$$\emptyset(Y) = \sum_{ij}\|y_i - y_j\|^2 w_{ij} \quad (8)$$

where the large weight w_{ij} in the cost function correspond to small distances between high-dimensional data point x_i and x_j.

3.3.4 Hessian Local Linear Embedding

Hessian Local Linear Embedding (HLLE) is a variant technique of LLE and is based on sparse matrix techniques [70]. It is an extension of the Linear Local Embedding (LLE) algorithm that aims to preserve the local geometric structure of the data. Compared to LLE, it typically produces outcomes of a significantly higher quality. Unfortunately, it is unsuitable for heavily sampled manifolds because of its expensive processing complexity.

The HLLE algorithm is based on the idea that the local geometric structure of the data can be approximated by the Hessian matrix, which measures the curvature of the data manifold. The Hessian matrix is used to compute a set of weights that are used to reconstruct each data point as a linear combination of its nearest neighbours.

The steps of the HLLE algorithm are as follows:

1. Find the k-nearest neighbours of each data point.
2. Estimate the Hessian matrix for each data point using its k-nearest neighbours.
3. Compute the weights that best reconstruct each data point as a linear combination of its k-nearest neighbours using the Hessian matrix.
4. Compute the low-dimensional embedding of the data using the weights and a sparse eigenvalue decomposition.

The resulting low-dimensional embedding is a smooth approximation of the data manifold that preserves the local geometric structure of the data. HLLE has been shown to be effective in a wide range of applications, including image analysis, robotics, and bioinformatics [71].

3.4 Summary of the Nonlinear Dimensionality Reduction Techniques

A summary of the characteristics of all the NLDRTs is presented in Table 1. Almost all the methods reviewed are nonparametric except the AE. Although all the techniques proved to be good choices, the AE's parametric nature made it an excellent method that can be utilised for the DR of nonlinear high-dimensional data. The AE has the propensity to accurately capture the topology of the manifold and tend to capture more global properties than other global methods.

Preserving local properties is the class of NLDRTs that attempts to maintain the data's local properties. The three classes of preserving local properties are (1) Reconstruction weights, (2) Neighbourhood graph Laplacian, (3) and Local tangent space. This section discusses all three techniques under the respective subclasses of preserving local properties techniques.

Table 1: Summary Of The Nonlinear Dimensionality Reduction Techniques

Technique	Type	Data Types	Deterministic	Objective	Cost Function
Kernel PCA	Nonparametric	Vectors	Yes	Distance preserving	Yes
MDS	Nonparametric	Distances	Yes	Distance preserving	Yes
LE	Nonparametric	Distance	Yes	Topology extraction	Yes
LLE	Nonparametric	Vectors	Yes	Manifold extraction	Yes
MVU	Nonparametric	Distances	Yes	Manifold extraction	Yes
Autoencoder	Parametric	Vectors	No	Information preserving	Yes
Isomap	Nonparametric	Distances	Yes	Manifold extraction	Yes
DP	Nonparametric	Distances	Yes	Manifold extraction	Yes
Hessian LLE	Nonparametric	Distances	Yes	Manifold extraction	Yes
t-SNE	Nonparametric	Distances	No	Manifold extraction	Yes
LLC	Nonparametric	Distances	Yes	Manifold extraction	Yes

4. CHALLENGES OF NONLINEAR DIMENSIONALITY REDUCTION TECHNIQUES

LDRTs have some limits and drawbacks despite the benefits. There are various challenges when transforming high-dimensional data into low-dimensional data. A significant problem that must be resolved is selecting a suitable technique based on the type of data [45]. In addition, it is essential to establish an effective method to attain the best level of accuracy while integrating the output of numerous DLRTs. The redundancy level of high-dimensional data must be determined since they have a variety of redundant features. Choosing the level of duplication and removing it without impairing low-dimensional mapping performance becomes difficult. Several NLDRTs offer this facility. Despite appearing redundant, some characteristics are essential for analysis and decision-making [72]. The low-dimensional data set that was necessary for the research has the potential to lose crucial information. Choosing an appropriate NLDRT in such circumstances can be challenging.

Several NLDRTs can improve data processing by using a few dimensions [73]. Important dimensions may occasionally not be chosen throughout the DR process, which harms the outcomes. For instance, omitting a trait with a lesser value but a significant impact on forecast accuracy will result in a poor choice. Most NLDRTs require pre-processing because they cannot operate with input data directly. Normalizing data is necessary to obtain accurate results [73].

Identifying the type of data and features utilized for analysis is a crucial consideration. Unfortunately, finding the pertinent and essential aspects has grown to be difficult. It requires domain knowledge and human skills to select ML models for classification and obtain the most essential characteristics for later processing. Another factor that renders the DR application unusable for applications requiring interpretation is the interpretation of outcomes [8].

After using DRTs, it is impossible to keep all of the high-dimensional data set information. Finding the most pertinent features for processing depends on the available data collection and the nature of the challenge. Due to the importance of most high-dimensional attributes for analysis, DR is occasionally impossible. The interconnectedness of the features is the cause. Dealing with the

interdependency of variables is another significant problem that must be solved. Numerous DRTs are useful for reducing noise and selecting only pertinent information for study [8]. Noise levels also impact DRT performance. Minor input modifications can influence results, which is the ultimate factor to be considered. For instance, values of a specific variable may be purposefully manipulated to extreme highs or lows. Inappropriate classification or predictions may result from this. Similarly, to this, choosing one incorrect feature can produce inaccurate results.

5. CONCLUSION

DRTs have attracted much attention over the past decades due to their application in various computing domains, such as face recognition, computer vision, pattern recognition, security, prediction, and classification. This paper reviews the most popular methods and techniques for NLDRTs. A thorough analysis of the pros and cons of NLDRTs used for data classification and visualisation was also discussed. The different applications where NLDRTs have been applied were also studied. The kind of issue and the underlying presuppositions of each approach determine which technique is most suitable. Comparative analysis of different methods helps in understanding the implementation of data analysis in a better way. The AE proved to be an excellent technique for DR of nonlinear high-dimensional data because it tended to capture the manifold's topology accurately and tend to capture more of the global properties than other global techniques.

NLDRTs are useful for analyzing high-dimensional data by projecting it onto a lower-dimensional space, but they come with several challenges. These challenges include computational complexity, overfitting, hyperparameter tuning, interpretability, limited applicability, and limited scalability. Nonlinear dimensionality reduction methods involve complex computations and can be time-consuming for large datasets. They may overfit the noise in the data rather than the underlying structure, making generalization to new data difficult. Finding optimal hyperparameters can be challenging and interpreting the resulting low-dimensional representations may be difficult. Additionally, some methods may not be suitable for all types of data, such as data with missing values or outliers, and they may not scale well to very large datasets.

Overall, NLDRTs can be a powerful tool for analysing high-dimensional data, but they require careful consideration and parameter tuning to achieve optimal results.

REFERENCES:

[1] Gandomi, A., Haider, M. (2015). Beyond the hype. Big data concepts, methods, and analytics. International journal of information management, 35(2), pp. 137-144.

[2] Han, K., Wang, Y., Zhang, C., Li, C., Xu, C. (2018, April). Autoencoder inspired unsupervised feature selection. In IEEE International Conference on Acoustics, Speech and signal processing (ICASSP), pp. 2941-2945.

[3] Huang, W., Martin, P., Zhuang, H. L. (2019). Machine-learning phase prediction of high-entropy alloys, Acta Materialia, 169, pp. 225-236.

[4] Shi, H., Li, H., Zhang, D., Cheng, C., Cao, X. (2018). An efficient feature generation approach based on deep learning and feature selection techniques for traffic classification. Computer Networks, 132, pp. 81-98.

[5] Sarker, I. H. (2021). Machine learning. Algorithms, real-world applications and research directions. SN Computer Science, 2(3), pp. 1-21.

[6] Folorunsho, O., Adegbola, I. A., Jimoh, R. G. (2022). An enhanced feature selection and classification model for network intrusion detection system using data mining techniques. Indian Journal of Computer Science and Engineering, 13(1), pp. 145-146.

[7] Wang, Y., Huang, H., Rudin, C., Shaposhnik, Y. (2021). Understanding how dimension reduction tools work: an empirical approach to deciphering t-SNE, UMAP, TriMAP, and PaCMAP for data visualization. The Journal of Machine Learning Research, 22(1), pp. 9129-9201.

[8] Ayesha, S., Hanif, M. K., Talib, R. (2020). Overview and comparative study of

dimensionality reduction techniques for high dimensional data. Information Fusion, 59, pp. 44-58.

[9] Jia, W., Sun, M., Lian, J., Hou, S. (2022). Feature dimensionality reduction: a review. Complex & Intelligent Systems, 8(3), 2663-2693.

[10] Hasan, B. M. S., Abdulazeez, A. M. (2021). A review of principal component analysis algorithm for dimensionality reduction. Journal of Soft Computing and Data Mining, 2(1), pp. 20-30.

[11] Singh, J., Azamfar, M., Li, F., Lee, J. (2020). A systematic review of machine learning algorithms for prognostics and health management of rolling element bearings. fundamentals, concepts and applications. Measurement Science and Technology. 32(1), pp. 1-52.

[12] Velliangiri, S., Alagumuthukrishnan, S. (2019). A review of dimensionality reduction techniques for efficient computation. Procedia Computer Science, 165, pp. 104-111.

[13] Negi, S., Kumar, Y., Mishra, V. M. (2016, September). Feature extraction and classification for EMG signals using linear discriminant analysis. In 2nd International Conference on Advances in Computing, Communication, & Automation (ICACCA)(Fall), pp. 1-6.

[14] Di Massa, G., Costanzo, S. (2021, November). Strategy for Dimensionality Reduction in Diagnostics Measurements of Huge Antenna Arrays. In IEEE Conference on Antenna Measurements & Applications (CAMA), pp. 181-182.

[15] Ali, M. U., Ahmed, S., Ferzund, J., Mehmood, A., Rehman, A. (2017). Using PCA and Factor Analysis for dimensionality reduction of Bio-informatics Data. arXiv preprint arXiv.1707.07189.

[16] Cheng, C. Y.; Hsu, C. C.; Chen, M. C. (2010). Adaptive kernel principal component analysis (KPCA) for monitoring small disturbances of nonlinear processes. Industrial & Engineering Chemistry Research, vol. 49(5), pp. 2254-2262.

[17] Lee, J. A., Verleysen, M. (2009). Quality assessment of dimensionality reduction. Rank-based criteria. Neurocomputing, 72(7-9), pp. 1431-1443.

[18] Elkhadir, Z., Chougdali, K., Benattou, M. (2016). Intrusion detection system using pca and kernel pca methods. In Proceedings of the Mediterranean Conference on Information & Communication Technologies 2015, pp. 489-497.

[19] Wang, Y., Sun, F., Li, X. (2020). Compound dimensionality reduction based multi-dynamic kernel principal component analysis monitoring method for batch process with large-scale data sets. Journal of Intelligent & Fuzzy Systems, 38(1), pp. 471-480.

[20] Bécavin, C., Tchitchek, N., Mintsa-Eya, C., Lesne, A., Benecke, A. (2011). Improving the efficiency of multidimensional scaling in the analysis of high-dimensional data using singular value decomposition. Bioinformatics, 27(10), pp.1413-1421.

[21] Zhu, W., Webb, Z. T., Mao, K., Romagnoli, J. (2019). A deep learning approach for process data visualization using t-distributed stochastic neighbor embedding. Industrial & Engineering Chemistry Research, 58(22), pp. 9564-9575.

[22] Shalini, T., Suganya, V. (2013). Clustering on High Dimensional Data Using Locally Linear Embedding (LLE) Techniques. Data Mining and Knowledge Engineering, 5(2), pp. 79-82.

[23] Wang, J., Wong, R. K., Lee, T. C. (2019). Locally linear embedding with additive noise. Pattern Recognition Letters, 123, 47-52.

[24] Sun, W., Halevy, A., Benedetto, J. J., Czaja, W., Liu, C., Wu, H., ..., Li, W. (2014). UL-Isomap based nonlinear dimensionality reduction for hyperspectral imagery classification. Journal of Photogrammetry and Remote Sensing, 89, pp. 25-36, 2014.

[25] Wang, Y., Yao, H., Zhao, S. (2016). Auto-encoder based dimensionality reduction. Neurocomputing, 184, pp. 232-242.

[26] Eiteneuer, B., Hranisavljevic, N., Niggemann, O. (2019, February). Dimensionality reduction and anomaly

detection for cpps data using autoencoder. In IEEE International Conference on Industrial Technology (ICIT), pp. 1286-1292.

[27] Van Der Maaten, L., Postma, E., Van den Herik, J. (2009). Dimensionality reduction. a comparative. J Mach Learn Res, 10, pp. 66-71.

[28] Charte, D., Charte, F., García, S., del Jesus, M. J., Herrera, F. (2018). A practical tutorial on autoencoders for nonlinear feature fusion. Taxonomy, models, software and guidelines. Information Fusion, 44, pp. 78-96.

[29] Sun, S., Cao, Z., Zhu, H., Zhao, J. (2019). A survey of optimization methods from a machine learning perspective. IEEE transactions on cybernetics, 50(8), pp. 3668-3681.

[30] Vocaturo, E., Zumpano, E., Veltri, P. (2018, December). Image pre-processing in computer vision systems for melanoma detection. In 2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 2117-2124.

[31] Arunachalam, S., Kshatriya, H. H., Meena, M. (2018). Identification of defects in fruits using digital image processing. International Journal of Computer Sciences and Engineering, 6(10), pp. 637-640.

[32] Mishra, V. K., Kumar, S., Shukla, N. (2017). Image acquisition and techniques to perform image acquisition. SAMRIDDHI. A Journal of Physical Sciences, Engineering and Technology, 9(1), pp. 21-24.

[33] Dash, S., Shakyawar, S. K., Sharma, M.; Kaushik, S. (2019). Big data in healthcare: management, analysis and future prospects. Journal of Big Data, 6(1), pp. 1-25.

[34] Olaiya, F., Adeyemo, A. B. (2012). Application of data mining techniques in weather prediction and climate change studies. International Journal of Information Engineering and Electronic Business, 4(1), pp. 51-59.

[35] Goswami, S., Chakrabarti, A. (2014). Feature selection. A practitioner view. International Journal of Information Technology and Computer Science, 6(11), pp. 66-62.

[36] Lv, Z., Song, H., Basanta-Val, P., Steed, A., Jo, M. (2017). Next-generation big data analytics. State of the art, challenges, and future research topic. IEEE Transactions on Industrial Informatics, vol. 13(4), pp. 1891-1899.

[37] LaFratta, C. N., Walt, D. R. (2008). Very high-density sensing arrays. Chemical reviews, 108(2), pp. 614-637.

[38] Zhang, R., Shim, B., Wu, W. (2021). Direction-of-Arrival Estimation for Large Antenna Arrays with Hybrid Analog and Digital Architecture. IEEE Transactions on Signal Processing, 70, pp. 72-88, 2021.

[39] McQuitty, S. (2018). The purposes of multivariate data analysis methods. an applied commentary. Journal of African Business, 19(1), pp 124-142.

[40] Selvan, C., Balasundaram, S. R. (2021). Data Analysis in Context-Based Statistical Modeling in Predictive Analytics. In Handbook of Research on Engineering, Business, and Healthcare Applications of Data Science and Analytics, pp. 96-114.

[41] Fan, J., Han, F., Liu, H. (2014). Challenges of big data analysis. National science review, 1(2), pp. 293-314.

[42] Orsenigo, C., & Vercellis, C. (2013). A comparative study of nonlinear manifold learning methods for cancer microarray data classification. Expert systems with Applications, 40(6), pp. 2189-2197.

[43] Xia, J., Ye, F., Chen, W., Wang, Y., Chen, W., Ma, Y., Tung, A. K. (2017). LDSScanner. Exploratory analysis of low-dimensional structures in high-dimensional datasets. IEEE transactions on visualization and computer graphics, 24(1), pp. 236-245, 2017.

[44] Sumithra, V., Surendran, S. (2015). A review of various linear and nonlinear dimensionality reduction techniques. International Journal of Computer Science and Information Technology, 6, pp. 2354-2360, 2015.

[45] Van der Maaten, L., Hinton, G. (2008). Visualizing data using t-SNE. Journal of machine learning research, 9(11), 2579-2605.

[46] Sarfraz, S., Koulakis, M., Seibold, C., Stiefelhagen, R. (2022). Visualization of high-dimensional data by pairwise fusion matrices using t-SNE. Symmetry, 11(1), pp. 107.

[47] Sarfraz, S., Koulakis, M., Seibold, C., Stiefelhagen, R. (2022). Hierarchical Nearest Neighbor Graph Embedding for Efficient Dimensionality Reduction. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 336-345.

[48] Dhalmahapatra, K., Shingade, R., Mahajan, H., Verma, A., Maiti, J. (2019). Decision support system for safety improvement. An approach using multiple correspondence analysis, t-SNE algorithm and K-means clustering. Computers & Industrial Engineering, 128, pp. 277-289.

[49] Chatzimparmpas, A., Martins, R. M., Kerren, A. (2020). t-visne. Interactive assessment and interpretation of t-sne projections. IEEE transactions on visualization and computer graphics, 26(8), 2696-2714, 2020.

[50] Anowar, F., Sadaoui, S., Selim, B. (2021). Conceptual and empirical comparison of dimensionality reduction algorithms (pca, kpca, lda, mds, svd, lle, isomap, le, ica, t-sne). Computer Science Review, 40, pp. 1-13.

[51] Huang, Y., Zha, X. F., Lee, J., Liu, C. (2013). Discriminant diffusion maps analysis. A robust manifold learner for dimensionality reduction and its applications in machine condition monitoring and fault diagnosis. Mechanical Systems and Signal Processing, 34(1-2), pp. 277-297.

[52] Farbman, Z., Fattal, R., Lischinski, D. (2010). Diffusion maps for edge-aware image editing. ACM Transactions on Graphics (TOG), 29(6), pp. 1-10.

[53] Ruiz, A., López-de-Teruel, P. E. (2001). Nonlinear kernel-based statistical pattern analysis. IEEE Transactions on Neural Networks, 12(1), pp.16-32.

[54] Li, Y., Zhou, R. G., Xu, R., Hu, W., Fan, P. (2020). Quantum algorithm for the nonlinear dimensionality Nguyen, L. H., Holmes, S. (2019). Ten quick tips for effective dimensionality reduction. PLoS computational biology, 15(6), e1006907

[55] Li, Y., Zhou, R. G., Xu, R., Hu, W., Fan, P. (2020). Quantum algorithm for the nonlinear dimensionality reduction with arbitrary kernel. Quantum Science and Technology, 6(1), 10-21.

[56] Xie, X., & Lam, K. M. (2006). Gabor-based kernel PCA with doubly nonlinear mapping for face recognition with a single face image. IEEE Transactions on Image Processing, 15(9), pp. 2481-2492.

[57] Kim, J., Calhoun, V. D., Shim, E., Lee, J. H. (2016). Deep neural network with weight sparsity control and pre-training extracts hierarchical features and enhances classification performance. Evidence from whole-brain resting-state functional connectivity patterns of schizophrenia. Neuroimage, 124, pp. 127-146.

[58] Sagheer, A., Kotb, M. (2019). Unsupervised pre-training of a deep LSTM-based stacked autoencoder for multivariate time series forecasting problems. Scientific reports, 9(1), pp. 1-6.

[59] Binbusayyis, A., & Vaiyapuri, T. (2021). Unsupervised deep learning approach for network intrusion detection combining convolutional autoencoder and one-class SVM. Applied Intelligence, 51(10), pp. 7094-7108.

[60] Xia, M., Li, T., Liu, L., Xu, L., & de Silva, C. W. (2017). Intelligent fault diagnosis approach with unsupervised feature learning by stacked denoising autoencoder. IET Science, Measurement & Technology, 11(6), pp. 687-695.

[61] Li, B., & Zhang, Y. (2011). Supervised locally linear embedding projection (SLLEP) for machinery fault diagnosis. Mechanical Systems and Signal Processing, 25(8), pp. 3125-3134.

[62] Roweis, S. T., & Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. science, 290(5500), pp. 2323-2326.

[63] Le, T. M., & Lauw, H. W. (2014, August). Semantic visualization for spherical representation. In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 1007-1016.

[64] Li, B., Zheng, C. H., & Huang, D. S. (2008). Locally linear discriminant embedding: An efficient method for face recognition. *Pattern Recognition*, *41*(12), pp. 3813-3821.

[65] Saul, L. K., & Roweis, S. T. (2003). Think globally, fit locally: unsupervised learning of low dimensional manifolds. *Journal of machine learning research*, *4*(Jun), 119-155.

[66] Verbeek, J. (2006). Learning nonlinear image manifolds by global alignment of local linear models. IEEE transactions on pattern analysis and machine intelligence, 28(8), pp. 1236-1250.

[67] Yuan, Z., He, Y., Yuan, L., Chen, P., Cheng, Z. (2020). An efficient feature extraction approach based on manifold learning for analogue circuits fault diagnosis. Analog Integrated Circuits and Signal Processing, 102(1), pp. 237-252.

[68] Pan, Y., Ge, S. S., Al Mamun, A. (2009). Weighted locally linear embedding for dimension reduction. Pattern Recognition, 42(5), pp.798-811.

[69] Tu, S. T., Chen, J. Y., Yang, W., & Sun, H. (2011). Laplacian eigenmaps-based polarimetric dimensionality reduction for SAR image classification. IEEE Transactions on Geoscience and Remote Sensing, 50(1), pp. 170-179.

[70] Liu, G., Li, X., Wang, C., Chen, Z., Chen, R., Qiu, R. C. (2022). Hessian Locally Linear Embedding of PMU Data for Efficient Fault Detection in Power Systems. IEEE Transactions on Instrumentation and Measurement, 71, pp. 1-4.

[71] Ge, S. S., He, H., & Shen, C. (2012). Geometrically local embedding in manifolds for dimension reduction. Pattern recognition, 45(4), pp. 1455-1470.

[72] Ray, P., Reddy, S. S., Banerjee, T. (2021). Various dimension reduction techniques for high dimensional data analysis. a review. Artificial Intelligence Review, 54(5), pp. 3473-3515.

[73] Asadi, S., Hadavandi, E., Mehmanpazir, F., Nakhostin, M. M. (2012). Hybridization of evolutionary Levenberg–Marquardt neural networks and data pre-processing for stock market prediction. Knowledge Based Systems 35, pp. 245-258.