Dimensionality Reduction: Challenges and Solutions

Noor Ahmad[1] Ali Bou Nassif[2,*]

[1] *University of Sharjah, Sharjah, UAE*
[2] *University of Sharjah, Sharjah, UAE*
[*] *Corresponding author. Email: anassif@sharjah.ac.ae*

ABSTRACT
The use of dimensionality reduction techniques is a keystone for analyzing and interpreting high dimensional data. These techniques gather several data features of interest, such as dynamical structure, input-output relationships, the correlation between data sets, covariance, etc. Dimensionality reduction entails mapping a set of high dimensional data features onto low dimensional data. Motivated by the lack of learning models' performance due to the high dimensionality data, this study encounters five distinct dimensionality reduction methods. Besides, a comparison between reduced dimensionality data and the original one using statistical and machine learning models is conducted thoroughly.

Keywords: *dimensionality reduction, GNB, KNN, LDA, PCA, random forest, SVD, SVM, t-SNE.*

1. INTRODUCTION

Despite the fact that Machine Learning Algorithms (MLAs) can handle large amounts of data [1], their efficiency degrades as the dimensionality of the data grows. [2]. Real-world data, such as speech signals, usually encounters a high dimensionality of features. A high number of features may slow down the induction process while giving similar results as obtained with a much smaller feature subset. To handle such real-world data effectively, data dimensionality requires to be decreased. Dimensionality Reduction (DR) is the makeover of high-dimensional data into a significant interpretation of diminished dimensionality. The fundamental dimensionality of data is the least number of parameters required to report for the observed attributes of the data [3].

Dimensionality reduction is essential in a variety of different areas since it diminishes the dimensionality and other unsought attributes of high-dimensional features [4], [5]. Usually, dimensionality reduction was performed using numerous statics methods such as Principal Components Analysis (PCA) [6], Linear Discriminant Analysis (LDA) [7], Singular Value Decomposition (SVD) [8], etc. Figure 1 displays a taxonomy of dimensionality reduction techniques along with their approaches. The taxonomy is subdivided into two main methods, which are reducing features dimension or selecting important features. In the first method, a combination of new reduced features is to be presented, known as, dimensionality reduction. Whereas in the second one, only the most important features are kept, known as features selection.

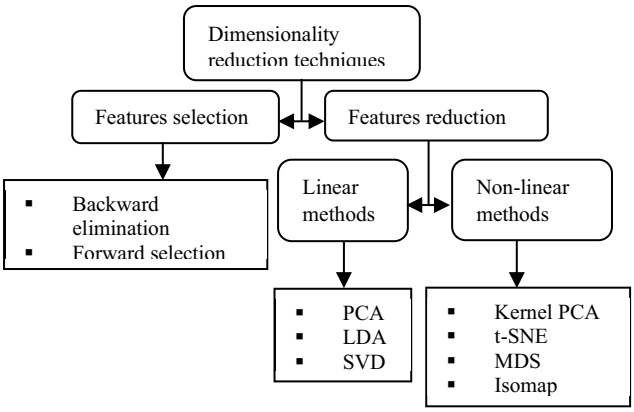

Figure 1. Dimensionality reduction taxonomy

The major motives for employing dimensionality reduction in machine learning are to enhance each of the prediction performance and the learning efficiency, to deliver faster prediction demanding less information on the original data, to decrease complexity and time of the learning outcomes and allow well understanding of the underlying procedure. This is very important when the input vector is large such as speech processing related problems [9], [10]. Lower data dimensions lead to less computing time and complexity with much less storage.

Additionally, fewer features help in the avoidance of overfitting [11]–[14].

Feature reducing and selection can be used to project data onto a lower dimensional space for subsequent clustering, visualization, and other experimental data analysis. These techniques can enhance classification accuracy by reducing estimation errors associated with finite sample size effects [15].

The rest of this paper is structured as follows: Section 2 discusses the techniques employed for dimensionality reduction. The dataset description appears in Section 3. Section 4 encounters several experiments along with the attained results. Section 5 states the concluding remarks.

2. DIMENSIONALITY REDUCTION TECHNIQUES

Techniques used to reduce the dimensions of a particular data are a vital solution to be encountered due to the enormous number of features that need to be eliminated cautiously. The subsequent is the explanation of the five techniques of some of these techniques. The following Figure 2 structure is the input and output targeted for each technique.

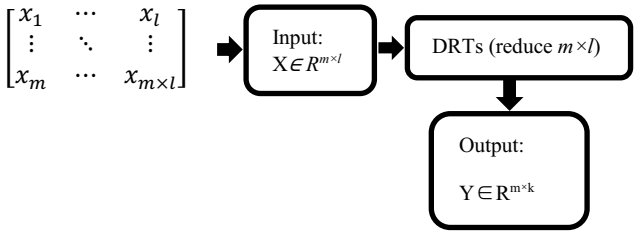

Figure 2. The general structure of DRTs steps

where, X is the input corpus (in high dimension), "m × l" is the dimension of the X. Also, Y is the output corpus (in low dimension), where "m × k" is the dimension after applying DRTs. "m" is number of data points.

2.1 Principal Components Analysis (PCA)

PCA is a multivariate statistical method that uses an orthogonal transformation and an effective method to improve computational time and accuracy. PCA describes as much variance as possible with the smallest number of variables, where an examination of the relationships between a group of variables. Additionally, to extract the essential information from the data and to convey this information as a set of other orthogonal variables called principal components. In mathematical phrases, n correlated random variables are transformed into a set of d ≤ n uncorrelated variables. These uncorrelated variables are linear combinations of the original variables and can be utilized to convey the data in a reduced form [6]. Assume that a dataset x(1), x(2), ..., x(m) has d dimension inputs. d-dimension data has to be reduced to k-dimension (k<<d) using PCA. The steps of PCA is described as follows [16]:

1) Standardization of the raw data: The raw data should have unit variance and zero mean.

$$x_j^i = \frac{x_j^i - \bar{x}_j}{\sigma_j} \quad \forall j \tag{1}$$

2) Compute the raw data's covariance matrix as follows:

$$\sum = \frac{1}{m} \sum_i^m (x_i)(x_i)^T, \sum \in R^{n*n} \tag{2}$$

3) Calculate the covariance matrix's eigenvector and eigenvalue as presented in Equation (4).

$$u^T \sum = \mu\lambda \tag{3}$$

$$U = \begin{bmatrix} | & | & | \\ u_1 & u_2 \dots & u_n \\ | & | & | \end{bmatrix}, \; u_i \epsilon R^n \tag{4}$$

4) The raw data must be translated onto a k-dimensional space: The top k eigenvectors of the covariance matrix are selected. These will be the data's new original foundation. Equation (5) shows how to calculate the equivalent vector.

$$x_i^{new} = \begin{bmatrix} u_1^T x^i \\ u_2^T x^i \\ \dots \\ \dots \\ u_k^T x^i \end{bmatrix} \epsilon R^k \tag{5}$$

Following that, if the raw data has n dimensions, it will be decreased to a new k-dimensional representation.

2.2 Linear Discriminant Analysis (LDA)

Linear Discriminant Analysis (LDA) is a linear, supervised Feature Extraction (FE) method. Despite this, some studies suggest that LDA can also be used as a linear classifier [23]. LDA establishes a new feature space in which to construct data with the purpose of increasing class separation. It derives k different independent features from a dataset's d independent characteristics that best split the classes (dependent features). As a result, the number of generated components is less than the number of classes. LDA constructs two scatter matrices at first, as seen in Equations (6) and (7) [34]:

1) an in-between-class matrix (SMb) that shows the distance between the means of each class.
2) A within-class matrix (SMw) computes the distance between each class's mean and the data within that class. Calculate the Eigen values and respective Eigen vectors of scatter matrices, then, to rank Eigen vectors by their values in descending order.
3) Build matrix W $(d \times k)$ with k top Eigen vectors.
4) Transform X using W to obtain the new subspace $Y = X.W$

$$SM_b = \sum_{k=1}^m N_k(\mu_k - \mu)(\mu_k - \mu)^T \tag{6}$$

$$SM_w = \sum_{k=1}^m \sum_{x=1}^n (x - \mu'_k)(x - \mu'_k)^T \tag{7}$$

where μ is the overall mean, μk is the mean and m is the number of classes. Also, Nk is and size of the corresponding classes and "μk" is the class's mean vector.

2.3 Singular Value Decomposition (SVD)

SVD allows a precise interpretation of any matrix, and likewise SVD is simple to eliminate the less important components of that interpretation to provide an estimated portrayal with desirable number of dimensions [17]. Assuming that an m×n matrix is defined by X. As per the following theory [18], the top k greatest singular values are picked.

1) U is a column-orthonormal matrix with $m \times k$ columns. The dot product of any two columns in this matrix is 0, and each column is a unit vector.

2) V is a column-orthonormal matrix with $n \times k$ columns. The rows of V^T that are orthonormal are represented by V in its transposed form. The columns are arranged in ascending order of importance.

3) S is a diagonal matrix with $k \times k$ elements. The number of elements that are not on the main diagonal is 0. The singular values of X are known as S elements.

4) If we divide a large matrix X into SVD components U, S, and V, these three matrices are also large to store [19]. Then,

$$Xm \times n = Um \times kSk \times k(Vn \times k)T \quad (8)$$

The SVD principle recovers a k-low dimension from the input matrix X, as shown in Equation (9), where U, S, and V^T are truncated forms of U, S, and V^T, respectively. Only the top k single values are saved in Y in this case.

$$Y = U_k \times S_k \times V_k^T \quad (9)$$

2.4 t-Distributed Stochastic Neighbor Embedding (t-SNE)

t-Distributed Stochastic Neighbor Embedding (t-SNE) is an unsupervised and nonlinear approach which represents low-dimension data from high-dimension data while preserving the substantial structure of the original data [20]. Mainly, The perception of how data is organized in a high-dimensional space is provided by t-SNE. In spite of receiving high performance, Dimensionality Reduction Algorithms (DRTs) are not frequently effective in visualizing high-dimensional data [20]. The t-SNE transforms high-dimensional Euclidean distances into conditional probabilities showing data similarity for each set using Stochastic Neighbor Embedding (SNE) [21]. The conditional probability $p_{a|b}$, defined in the equation below, exemplifies the resemblance of data x_a to data x_b [20]:

$$p_{a|b} = \frac{exp\frac{-\|x_b - x_a\|^2}{2\sigma^2}}{\sum_{a \neq k}\frac{-\|x_k - x_a\|^2}{2\sigma^2}} \quad (10)$$

Equation (10) calculates the distance between two data points $x_a$and x_b using a Gaussian distribution over x_b and a given variance of σ^2, where it differs for each data set and is chosen so that data from dense areas have smaller variance than data from sparse areas [20]. Then, a "Student t-distribution" is utilized as a substitute of utilizing the Gaussian distribution with one degree of freedom, close to the Cauchy distribution, is used to get the second set of probabilities ($Q_{a|b}$) in the low dimension space [22]. If the low dimension data ya and y_b are precisely mapped from the high dimension data x_a and x_b, then the similarity between $p_{a|b}$ and $Q_{a|b}$ happen to be equivalent. As a result, from low to high dimensional spaces, t-SNE reduces the difference between these two probabilities. As illustrated below, this difference is calculated by maximizing the cost function (φ) of the sum of Kullback–Leibler differences [22]:

$$\varphi = \sum_a \sum_b p_{a|b}\, log\frac{p_{a|b}}{Q_{a|b}} \quad (11)$$

In short, the t-SNE technique can be summarized as the following steps:

1) Apply SNE to X to calculate the conditional probabilities $p_{a|b}$ and $Q_{a|b}$.
2) Map X to Y by minimizing the difference between $p_{a|b}$ and $Q_{a|b}$ based on the cost function φ.

2.5 Independent Component Analysis (ICA)

ICA technique is a supervised and linear feature extraction method that produces statistically independent new features by decreasing the second order and higher order dependencies in a dataset [23]. The difference between ICA and other FEAs is that ICA looks for non-Gaussian, statistically independent features. PCA, for instance, aims for the best representations of the data, while ICA looks for the most independent (from one another) representations. At the start, ICA decomposes the data X as follows [24]:

$$X \rightarrow A.S \quad (12)$$

where S is the basis coefficient and A is the mixing matrix (the features are as independent as possible). ICA generates data Y by choosing the top k independent components from a data set to generate k dimensions:

$$Y = A_k.S_k \quad (13)$$

The components can be attained in particular sequence and scales [24]. ICA considered as a unique circumstance of the "blind source separation" issue recognized in the signal processing arena [25], in which The separation of original signals from mixed data with hardly any information about the source signals or the mixing process is the emphasis of ICA. It's important to keep in mind that the "Scikitlearn" tool makes use of "FastICA" to make ICA computationally and memory efficient. As a result, the following actions need be followed to achieve ICA:

1) Decompose X to A and S.
2) Select top k independent components.
3) Build Y by using k components.

The key concepts of Dimensionality reduction techniques are summarized in Table 1.

Table 1. Conceptual comparison of dimensionality reduction techniques

Techniques	*Goal and Strength*	*Weaknesses*	*Computational complexity*
PCA	Maximize the variance with low noise sensitivity	Limited to linear projection	$O(d^2n + n^3)$
LDA	Maximize class separation	Suffer from class singularity issue	$O(d^2n)$, n>d; $O(d^3)$, d>n
SVD	Minimum construction error	Not suitable for nonlinear data	$O(d^2n + n^3)$
t-SNE	Preserve local structure	Provide only 2 or 3 features	$O(n^2)$
ICA	Maximize statistical independence	Require high training time	O[2di(d+1)n]

3. DATASET DESCRIPTION

The assessment of the dimensionality reduction techniques performance is accomplished using the ECG200 corpus that arrives with 96 features. The database was structured by Olszewski as part of his thesis in [26]. The ECG200 was examined by domain experts and interpreted into two classes: ordinary heartbeat or Abnormal (Myocardial Infarction), as shown in Figure 3. Nevertheless, it includes only 200 observations in which 67 are Abnormal and 133 are Normal. Figure 4 depicts a histogram of certain features chosen at random to depict their distribution. As shown in this figure, the features have distinct value distributions. Symmetric data, e.g., feature of column 29, have roughly similar shape on both sides. Column 41's feature shows a multi-modal histogram, which indicates there are two or more peaks in this feature (local maxima). The mean is greater than the median if a histogram is skewed to the right (e.g., feature of column 95). This occurs when skewed-right data has some high values, driving the mean upward. Whereas, column 50 exhibits a skewed left histogram, with the mean smaller than the median. In this case, the presence of relatively lower values lowers the mean for this attribute. As a result, this dataset has a decent mix of features associated with a wide range of distributions.

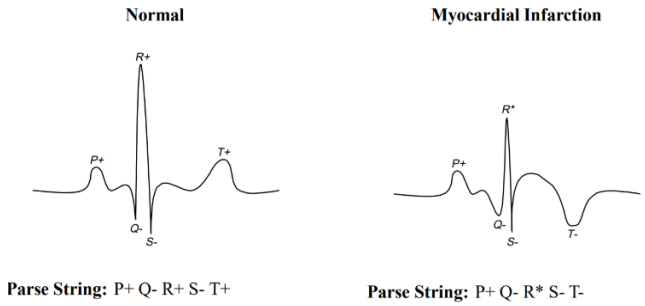

Figure 3. The classes of heartbeat

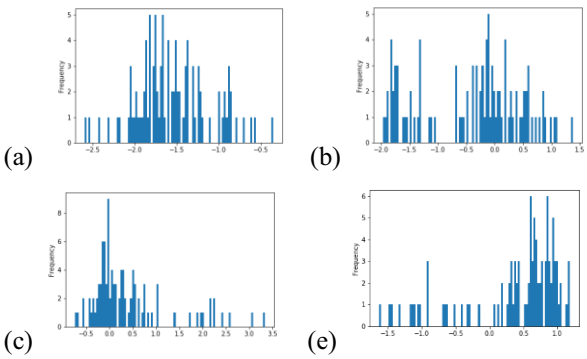

Figure 4. (a) Symmetric histogram, (b) Multimodal histogram, (c) Skewed-right histogram, and (d) Skewed-left histogram

4. RESULTS AND DISCUSSIONS

Evaluating the quality of the produced dataset will be accomplished by comparing the correlation (in the matter of p-value), F1-score, classification accuracy, precision, recall, ROC curve, and run-time metrics. The comparisons are performed between the original dataset and the reduced one.

The p-value used to calculate the statistical significance of an examination and is based on a predetermined significance level. Table 2 demonstrates the p-values for each DRTs. If the achieved p-value is less than 0.05, then the result is statistically significant [26]. After utilizing DRTs, lower p-values than the values generated with the original dataset have been acquired. For the evaluation, the LDA's accuracy was not considered as it yields to only one feature for this corpus. Therefore, the new reduced dataset is of superior quality to the original database.

The run-time in milliseconds (ms) of the six transformed databases and the original is illustrated in Table 3. SVM classifier with RBF kernel and other classifiers are trained and tested on the new feature spaces. Table 4 shows many MLAs based on the F1-score before and after the DRTs are applied. For instance, KNN returned an F1-score of 92 percent with a speed of 195.3 (ms) on the original data with 96 features. Followed by SVM, an 89% of F1-score has been achieved with a speed of 200 (ms). By employing SVM, there is a difference of 4% among the original and the top decreased feature space (with PCA), which is substantial in the medical

field. It is evident that the classifier using the reduced features space outperformed the original one. Besides, the KNN classifier with the PCA approach remarked the fastest classification time with only 3.2 ms, where SVD comes in second with seven (ms).

Table 2. Best P-value of initial and reduced databases

Dataset	*Dimension*	*p-value*
Initial	96	0.003
PCA	40	$1.31e^{-11}$
LDA	1	$7.8e^{-76}$
SVD	30	$6.7e^{-12}$
t-SNE	2	$5.56e^{-20}$
ICA	26	0.000007

Table 3. Classification performance of decreased and original databases using multiple MLAs in terms of run-time (ms)

Dataset	*Dimension*	*SVM*	*KNN*	*GNB*	*random forest*
Original	96	200.1	195.3	180.0	190.5
PCA	40	3.600	3.200	3.000	3.500
LDA	01	0.010	0.008	0.007	0.010
SVD	30	7.000	7.000	5.000	6.000
t-SNE	02	156.5	155.0	145.7	150.9
ICA	26	13.00	12.80	10.74	12.50

Table 4. performance of original and reduced datasets using multiple MLAs in terms of F1-score

Dataset	*Dimension*	*SVM*	*KNN*	*GNB*	*random forest*
Original	96	0.89	0.92	0.81	0.86
PCA	40	0.95	0.93	0.88	0.87
LDA	01	0.98	0.97	0.96	0.97
SVD	30	0.94	0.94	0.93	0.87
t-SNE	02	0.89	0.94	0.85	0.89
ICA	26	0.93	0.93	0.85	0.90

The classification performance of original and reduced datasets using multiple MLAs in terms of accuracy is shown in Table 5. In this table, the KNN classifier performed the best among the other MLAs. However, the SVM classifier surpassed the other classifiers when DRTs were applied except for t-SNE. Further important metrics can be illustrated through precision and recall. Table 6 the precision and recall of original and reduced datasets operating multiple MLAs. Results show that SVM (machine learning model) and random forest (statistical model) have roughly similar results. After analyzing and evaluating the tables, a decent ranking based on classification and data quality (correlation) performance are both met. In terms of data quality, PCA and SVD occupied first place, followed by ICA and t-SNE, respectively. In terms of performance, ICA remarked first place followed by PCA, SVD, and t-SNE, respectively.

Table 5. Classification performance of original and reduced datasets using multiple MLAs in terms of accuracy

Dataset	*Dimension*	*SVM*	*KNN*	*GNB*	*random forest*
Original	96	0.89	0.92	0.82	0.86
PCA	40	0.95	0.93	0.94	0.93
LDA	01	0.98	0.96	0.96	0.97
SVD	30	0.95	0.93	0.94	0.87
t-SNE	02	0.89	0.94	0.83	0.89
ICA	26	0.95	0.94	0.85	0.89

Table 6. Precision and recall of original and reduced datasets using multiple MLAs

Dataset	*P/R*	*SVM*	*KNN*	*GNB*	*random forest*
Original	P	0.88	0.92	0.79	0.87
	R	0.88	0.86	0.80	0.83
PCA	P	0.93	0.92	0.88	0.88
	R	0.93	0.93	0.88	0.86
LDA	P	0.98	0.98	0.96	0.97
	R	0.98	0.97	0.95	0.96
SVD	P	0.95	0.93	0.86	0.89
	R	0.95	0.91	0.86	0.83
t-SNE	P	0.89	0.94	0.84	0.88
	R	0.88	0.94	0.82	0.84
ICA	P	0.94	0.94	0.85	0.87
	R	0.94	0.93	0.85	0.84

**P: precision, R: recall*

One more significant metric called the ROC curve has been employed in the evaluation. ROC curve is short for Receiver Operating Characteristic Curve. ROC signifies the graphical scheme that characterizes the analytical ability of a classifier structure as its discernment threshold is altered. An example of ROC curves outputs on the GNB classifier has been visualized in Figure 5 in four main conditions: using each of t-SNE, ICA, PCA, and SVD, and without using any DRTs (original).

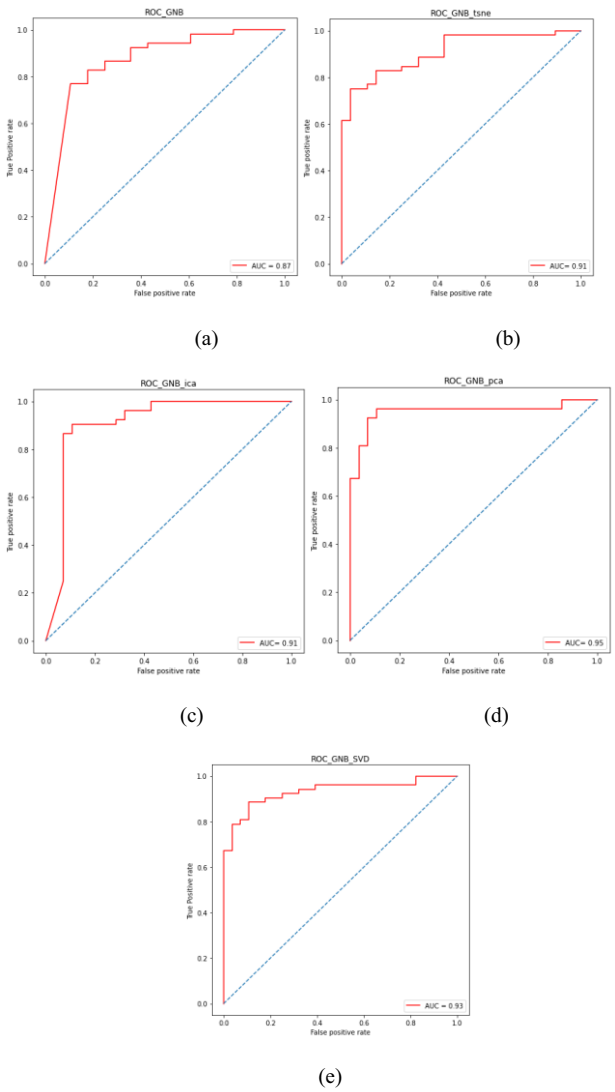

Figure 5. Applying DRTs on GNB classifier (a) without DRTs, (b) t-SNE, (c) ICA, (d) PCA, and (e) SVD

5. CONCLUDING REMARKS

The database for MLAs should be of great quality and should note trivial or redundant information; else, performance will be unreliable. Due to that, this article presents five distinct Dimensionality Reduction Techniques (DRTs). Moreover, a thorough examination was performed via multiple assessments including the comparison between the MLAs' performance before and after applying DRTs. The performance of each MLAs using each of PCA, LDA, SVD, t-SNE, ICA was conducted. Results are empirically evaluated based on the p-value, precision, recall, classification accuracy, F1-score, ROC curve, and run-time metrics. Two main interpretations are remarked, where data quality and classification accuracy improved when DRTs were used. Besides that, in the majority of situations, nonlinear DRTs performed better than linear ones.

One of this paper's limitations is the utilization of one dataset with no parameters optimization. Besides, more DRTs should be tested and compared with each other.

For future study objectives, an exploration of the performance of deep learning with DRTs on high dimensional databases will be performed along with parameter optimization. As well, DRTs are to be explored on multiple complex databases, such as multi-label data and multi-dimensional time-series.

REFERENCES

[1] F. Anowar and S. Sadaoui, “Incremental Neural-Network Learning for Big Fraud Data,” in *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2020, pp. 3551–3557, doi: 10.1109/SMC42975.2020.9283136.

[2] L. J. P. Van Der Maaten, E. O. Postma, and H. J. Van Den Herik, “Dimensionality Reduction: A Comparative Review,” *J. Mach. Learn. Res.*, vol. 10, pp. 1–41, 2009, doi: 10.1080/13506280444000102.

[3] K. Fukunaga, *Introduction to Statistical Pattern Recognition*. San Diego, CA, USA, 1990.

[4] L. O. Jimenez and D. A. Landgrebe, “Supervised classification in high-dimensional space: geometrical, statistical, and asymptotical properties of multivariate data,” *IEEE Trans. Syst. Man, Cybern. Part C (Applications Rev.*, vol. 28, no. 1, pp. 39–54, 1998, doi: 10.1109/5326.661089.

[5] F. Salo, A. B. Nassif, and A. Essex, “Dimensionality reduction with IG-PCA and ensemble classifier for network intrusion detection,” *Comput. Networks*, vol. 148, pp. 164–175, Jan. 2019, Accessed: Sep. 03, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1389128618303037.

[6] M. Partridge and C. Rafael, “Fast dimensionality reduction and simple PCA,” *Intell. Data Anal.*, vol. 2, no. 3, pp. 292–298, 1998, doi: 10.3233/IDA-1998-2304.

[7] P. Switzer, “Extensions of linear discriminant analysis for statistical classification of remotely sensed satellite imagery,” *J. Int. Assoc. Math. Geol.*, vol. 12, no. 4, pp. 367–376, 1980, doi: 10.1007/BF01029421.

[8] K. V Ravi Kanth, D. Agrawal, A. El Abbadi, and A. Singh, “Dimensionality Reduction for Similarity Searching in Dynamic Databases,” *Comput. Vis. Image Underst.*, vol. 75, no. 1, pp. 59–72, 1999, doi: https://doi.org/10.1006/cviu.1999.0762.

[9] I. Shahin, S. Hamsa, “Novel cascaded Gaussian mixture model-deep neural network classifier

for speaker identification in emotional talking environments," *Neural Comput. Appl.*, pp. 1–13, Oct. 2018, doi: 10.1007/s00521-018-3760-2.

[10] A. B. Nassif, I. Shahin, S. Hamsa, N. Nemmour, and K. Hirose, "CASA-Based Speaker Identification Using Cascaded GMM-CNN Classifier in Noisy and Emotional Talking Conditions," *Appl. Soft Comput.*, vol. 103, pp. 1–24, 2021, doi: 10.1016/j.asoc.2021.107141.

[11] S. Khalid, T. Khalil, and S. Nasreen, "A survey of feature selection and feature extraction techniques in machine learning," in *2014 Science and Information Conference*, 2014, pp. 372–378, doi: 10.1109/SAI.2014.6918213.

[12] A. B. Nassif, M. Azzeh, L. F. Capretz, and D. Ho, "A comparison between decision trees and decision tree forest models for software development effort estimation," in *2013 3rd International Conference on Communications and Information Technology, ICCIT 2013*, 2013, pp. 220–224, doi: 10.1109/ICCITechnology.2013.6579553.

[13] M. Azzeh "Analogy-based effort estimation: a new method to discover set of analogies from dataset characteristics," *IET Softw.*, vol. 9, no. 2, pp. 39–50, 2015, doi: 10.1049/iet-sen.2013.0165.

[14] M. Azzeh, S. Banitaan, and F. Almasalha, "Pareto efficient multi-objective optimization for local tuning of analogy-based estimation," *Neural Comput. Appl.*, vol. 27, no. 8, pp. 2241–2265, 2016, doi: 10.1007/s00521-015-2004-y.

[15] A. K. Jain and B. Chandrasekaran, "39 Dimensionality and sample size considerations in pattern recognition practice," in *Classification Pattern Recognition and Reduction of Dimensionality*, vol. 2, Elsevier, 1982, pp. 835–855.

[16] H. Abdi and L. J. Williams, "Principal Component Analysis," *Wiley Interdiscip. Rev. Comput. Stat.*, vol. 4, no. 2, pp. 433–459, 2010.

[17] A. Rajaraman and J. D. Ullman., *Mining of massive datasets*. Cambridge University Press, 2011.

[18] A. G. Akritas and G. I. Malaschonok, "Applications of singular-value decomposition (SVD)," *Math. Comput. Simul.*, vol. 67, no. 1, pp. 15–31, 2004, doi: https://doi.org/10.1016/j.matcom.2004.05.005.

[19] H. Abdi, *Singular value decomposition (SVD) and generalized singular value decomposition*. 2007.

[20] A. Van der Maaten and G. Hinton, "Visualizing Data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 187–202, 2008, doi: 10.1007/s10479-011-0841-3.

[21] B. Melit Devassy and S. George, "Dimensionality reduction and visualisation of hyperspectral ink data using t-SNE," *Forensic Sci. Int.*, vol. 311, p. 110194, 2020, doi: https://doi.org/10.1016/j.forsciint.2020.110194.

[22] F. H. M. Oliveira, A. R. P. Machado, and A. O. Andrade, "On the Use of *t*-Distributed Stochastic Neighbor Embedding for Data Visualization and Classification of Individuals with Parkinson's Disease," *Comput. Math. Methods Med.*, vol. 2018, p. 8019232, 2018, doi: 10.1155/2018/8019232.

[23] L. J. Cao, K. S. Chua, W. K. Chong, H. P. Lee, and Q. M. Gu, "A comparison of PCA, KPCA and ICA for dimensionality reduction in support vector machine," *Neurocomputing*, vol. 55, no. 1, pp. 321–336, 2003, doi: https://doi.org/10.1016/S0925-2312(03)00433-8.

[24] C. O. S. Sorzano, J. Vargas, and A. P. Montano, "A survey of dimensionality reduction techniques," pp. 1–35, 2014.

[25] A. Tharwat, "Independent component analysis: An introduction," *Appl. Comput. Informatics*, vol. 17, no. 2, pp. 222–249, Jan. 2021, doi: 10.1016/j.aci.2018.08.006.

[26] R. T. Olszewski, "Generalized feature extraction for structural pattern recognition in time -series data," Carnegie Mellon University, Ann Arbor, 2001.