Enhancing K-Nearest Neighbors with Dimensionality Reduction: An Impact on Recognition Time and Accuracy

Priyadarshan Dhabe[1], Harshita Bhagat[2], Parth More[3], Vaishali More[4], Chinmay Saraf[5] and Sarthak Khare[6]
[1-6]Department of Information Technology, Vishwakarma Institute of Technology, Pune
Email: {priyadarshan.dhabe, harshita.bhagat21, parth.more21, vaishali.more21, chinmay.saraf211, khare.sarthak211}@vit.edu

***Abstract*—KNN [1] is the simplest and popular classifier used for pattern recognition tasks and predictions across various application domains. However, certain limitations hinder its efficiency. These include the higher computational cost due to excessive recognition time and more memory space. KNN requires significant computer memory as the complete training dataset needs to be stored in the computer memory. Thus, it is less suitable for larger training and testing datasets with higher dimensionality. KNN requires time and memory proportional to the size of training and testing datasets and their dimensionality. To reduce this handicap of KNN, in this paper, we propose a simple statistical method to reduce the dimensionality of features used in training and testing datasets. The proposed approach helps in reducing recognition time and memory space both. We did experimentation by using 3 datasets available on Kaggle [2] namely the breast cancer dataset [3], Wine quality dataset [4] and the diabetes dataset [5]. As per our experimentation, we observed, on an average, 27% reduction in recognition time as compared to using all the pattern features. We also found 18% to 30% reduction in number of features, without reduction in accuracy. In fact, for breast cancer and wine quality dataset, there is a slight increase in recognition accuracy too. Thus, we recommend this approach to use KNN for larger datasets with high dimensionality also.**

***Index Terms*—KNN, Dimensionality Reduction, Recognition Time, Classification and Recognition accuracy.**

I. INTRODUCTION

K-Nearest Neighbor (KNN) is a popular algorithm for making predictions in diverse domains. It depends on the proximity of the points to classify or predict new instances. However, KNN poses significant challenges such as high computational costs, due to the need to compare data points with all others, resulting in extended recognition times. An increase in the number of features makes generalization of the data difficult. As discussed in the paper [6] the dimensionality curse affects the performance of the model for high-dimensional data. Use of KNN is a costly affair, if the numbers of features are more, as higher dimensional space needs more computations to calculate the distance of w.r.t. neighboring data points considering all the dimensions. It results in high recognition times. Additionally, its reliance on storing and referencing a substantial amount of training data can lead to substantial memory requirements, making it less practical/applicable for huge datasets. These challenges make KNN less efficient in scenarios with massive datasets and real-time processing demands. Furthermore, reducing feature dimensionality can also lead to improve model interpretability, which is important for understanding the decision-making process of KNN. This can be particularly advantageous in domains where transparency and explainability are important considerations. As a result, researchers have turned their attention to techniques for reducing dataset size, achieved by trimming both samples and the features.

This paper presents a statistical approach of dimensionality reduction based on the concept of standard deviation. This dimensionality reduction approach will be helpful to reduce the effect curse of dimensionality which occurs for the data with more than 10 dimensions [7]. Features must be removed in such a way that removal should not affect the classification as well as recognition accuracy of model. Alternative methods for reducing dimensionality, distinct from Principal Component Analysis (PCA [8]), which try to maximize the variance and find linear relationships of the original features [8], the proposed spread-based method considers all features independently, this non-linear method does not consider correlations. This method directly removes the features with low spread. Features with extremely less spread do not affect the accuracy of the model significantly. Striking the right balance between dimensionality reduction and model performance is crucial. We verified the recognition and classification accuracy after the reduction of features by applying KNN model to the reduced feature dataset.

A. Working of KNN

KNN is a supervised machine learning technique employed for classification and regression purposes. Its operation involves determining the 'k' closest neighbors within the training dataset for a new data point and determining its label or value through the majority vote of these neighbors.

1. **Data Representation:** For a given dataset with training points belonging to three classes viz. class 1, 2 and 3, are shown in Fig. 1. We also depict a new unlabelled test data point P in Fig. 1.
2. **Distance Calculation:** KNN works by calculating distances of this new point P with all the training points, based on a chosen distance metric d (Manhattan distance, Euclidean distance, etc.). An Euclidean distance can be computed using (1).

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(y_i - x_i)^2} \qquad (1)$$

here, x and y are two points in n-dimensional Euclidean space

3. **Define K:** Further, we select a user defined neighborhood size $K > 0$ as nearest neighbors of P depending on the calculated distances.
4. **Prediction:** KNN can be used for classification: The new data point P is assigned the majority class among its K neighbors in classification, and regression: The new data point is assigned the average value of the K neighbors' target variables.

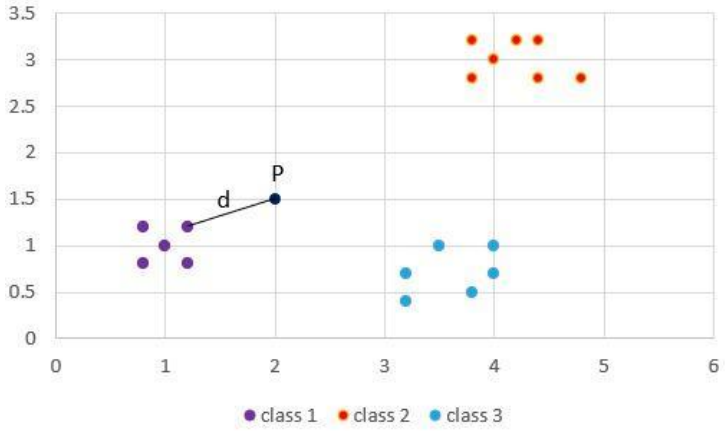

Figure 3. Scatter Plot of 2D patterns and working of KNN

B. Drawbacks of KNN

Following are the drawbacks of the original KNN.

1. KNN requires more recall time for larger data sets with higher dimensionality.
2. It is Sensitive to noise and missing data also requires feature scaling.
3. It requires high memory space since it needs to store all the training points.
4. Choosing an optimal value for 'k': higher value of 'k' leads to more neighbors, causing a smoothing effect and ignoring local variations and trends in data. But, lesser values of 'k' might lead to the over fitting problems, causing failure to classify unseen data accurately.

C. Derived motivation for this work

To improve the efficiency of KNN algorithm, we focus on reducing computational complexity. We integrated dimensionality reduction technique to enhance recognition time with the same accuracy, thus making KNN more suitable for real-time applications even with larger datasets with high dimension.

D. Organization of the paper

The research is divided into six sections. It begins with an overview of studies related to the current work in Section 2, followed by the proposed methodology in Section 3. The results are discussed in Section 4, and the conclusion is provided in Section 5. In Section 6 we discuss the future scope of the research.

II. Literature Survey

Grzegorz Borowik et al. [9], used logic synthesis to reduce hardware complexity and optimize system implementation. In this paper, the authors highlight the importance of dimensionality reduction in the compact representation of the object. It gives a comparison of different programs like RSES, jMAF, and Weka to check the effectiveness of dimensionality reduction [9]. Yinglei Song et al. [10] addressed the challenge of dimensionality reduction; authors emphasized preserving global and local variables. They developed a quadratic measure to describe crucial local features and formulate an optimization problem with multiple objectives to achieve supervised dimensionality reduction. In sum, they have used a heuristic approach for a better solution. This research can be extended to get a more optimal solution and higher accuracy [10].
Mateus Espadoto et al. [11], have presented their survey involving Dimension Reduction (DR) techniques (44 in total) including a study showing how quality is dependent on the parameters of the projection algorithm. They also propose traits to characterize datasets and projection techniques, aiming to reflect the factors that non-expert users take into account when selecting a technique. Which makes the paper that much more valuable from a practical standpoint [11]. Dunja Mladenić delves into dimensionality reduction in machine learning, where the goal is to transform high-dimensional feature spaces. It outlines common techniques for selecting feature subsets, with a focus on their application to text data. The paper demonstrates the performance of these methods in real-world document categorization tasks [12]. S Sivaranjani et al. utilized machine learning to predict diabetes-related diseases, emphasizing their significance in the context of global and Indian healthcare. For dimensionality reduction this research employed PCA and feature selection [13]. Shaeela Ayesha et al. provide a review of dimensionality reduction techniques and their variants, particularly focusing on linear and non-linear techniques as applied to text, image, and signal data. The authors have observed how Dimensionality Reduction Techniques (DRTs) have been utilized in diverse data structures and application contexts [14]. The "curse of dimensionality" is a problem that affects computation and memory space, as well as the accuracy and performance of problem-solving. Feature reduction is a challenging research field that aims to find the best mapping for optimal low-dimensional data, and the survey done by Weikuan Jia et al discusses various methods in terms of Feature selection and Feature extraction. All the methods from linear feature extraction to local linear embeddings are covered in feature extraction.The survey also discussed supervised and unsupervised ways of dealing with high-dimensional data [15]. Three feature selection algorithms, FCBF, FCBF#, and FCFBiP, were compared, and FCFBiP was found to be the most efficient. Feature selection has many benefits, including preventing overfitting, reducing storage requirements, and improving computational cost. Wrapper methods and embedded methods are commonly used for feature selection. FCBF and FCBF# are effective but consume time due to training intensively, this is overcome by FCFBiP as it reduces elapsed time [16]. KNN algorithm is a highly utilized classification algorithm due to its straightforward implementation and effectiveness across various fields. However, it faces limitations, such as substantial memory demands and

elevated computational complexity. This paper introduces an approach combining dynamic selection, attribute weighting, and distance weighting techniques [17]. In integrating KNN classifiers with resampling strategies, the research proposes a creative method to get around the diversity constraints. Using multimodal perturbation, this method preserves the accuracy of component classifiers while enhancing their diversity. In order to lessen KNN's sensitivity to input attributes, the authors adopted a weighted heterogeneous distance measure (WHDM). They use evidence theory and WHDM to create a progressive KNN classifier. Then, in order to create an ensemble classifier, they combined random subspace, attribute reduction, and bagging techniques in a novel algorithm known as RRSB (Reduced Random Subspace-based Bagging). Experiments on a range of datasets confirm that RRSB is a useful tool for improving kNN ensembling and greatly advancing machine learning ensemble methods. [18].

The research papers highlight the importance of simplifying complex data for better computer systems and machine learning. They explore methods like logic synthesis and quadratic measures to reduce data dimensions. The papers emphasize using practical strategies, like combining classifiers and introducing weighted distance metrics, to improve machine learning effectiveness. Overall, the focus is on making data manageable, enhancing system performance, and finding practical solutions for diverse applications.

III. Proposed Methodology

This section provides the methodology of the proposed system to convert higher dimensional datasets into lower dimensional datasets by discarding some features such that accuracy of the model should not be affected by their reduction. Model training is described below.

Assume P = { $x_1, x_2, \ldots, x_n$ } a n dimensional pattern, with x_i, where $i = 1, 2, \ldots, n$ are statistical features for $n > 0$ and $i \geq 1$. In our approach, we will try to find m important features, $m \ll n$, $m > 0$ such that by using m features only, we will get comparable accuracy with reduction in recognition time. We proposed a statistical method based on the variance of the feature. The variance of the sample is calculated using (2). We delete k^{th} feature x_k if variance in k^{th} feature is less, compared to other features. We used this modified dataset without k^{th} feature and done training and testing, accuracy is noted for the corresponding dataset.

$$\sigma^2 = \frac{\Sigma(x_i - \bar{x})^2}{n-1} \quad (2)$$

here, σ^2: variance of dataset, x_i: observed value,$\bar{x}$: mean value of n observations and n: total number of observations.

In further iterations, more number of features with lesser variance compared to other features are deleted and above steps are repeated till a significant number of features are reduced and the optimum result is obtained. In each step classification, recognition accuracy and time taken to predict the label are observed. At one stage we arrive at an optimum solution where classification and recognition accuracies are comparable or same and testing time is decreasing with a significant amount, this represents the solution for selecting the features to train the KNN model.

IV. Results and Discussion

We experimented with our model with breast cancer dataset, Wine quality dataset, diabetes dataset and dry beans dataset. Features from these datasets are reduced using standard deviation for training the KNN model as discussed above in the methodology section. For all datasets, 80% of samples are used for training and 20% of samples are used for testing. Classification accuracy is the performance of a model for training dataset and Recognition accuracy is performance for testing dataset. The time taken to recognize class labels in a testing dataset is calculated by taking the average time after 100 epochs to get more accurate time. Results obtained from the implemented model are discussed in this section.

A. Breast Cancer Wisconsin

This dataset contains a total of 31 independent variables and one output variable with 2 classes. Figure [2] shows percentage decrease in recognition time w.r.t percentage feature reduction, recognition time decreases when we decrease the number of features from the dataset as calculations for those features in the prediction also decrease.

Figure [3] shows deviation in classification and recognition accuracy with reduction in features. The blue and orange line in the graph represents the effect of feature reduction on classification and recognition accuracy

respectively. Generally, when features are reduced in the training then the model may lose some important information hence leading to a less accurate model, but when features are reduced based on spread, one set of features gives the best accuracy without affecting much of the original accuracy which we call here as optimal solution. The main goal of this study is to reduce the recognition time of the model. Table [1] shows, when there is no feature reduction, accuracy of classification is 97.48% and as we increase percentage reduction in features accuracy decreases. For reduction percentage 3.33 to 13.33 decrease in testing time is high i.e nearly 55% but classification accuracy and the difference between classification and recognition accuracy is more and for more than 30% reduction the stable condition for accuracy and time is not achieved as we get better results for the highlighted row observation. In figure 3, for 30% reduction optimum solution is achieved where original accuracy with all features is achieved and recognition accuracy is increased, we were able to decrease recognition time by 30.25%. And more than 30% reduction in features is also leading to non-optimal solutions.

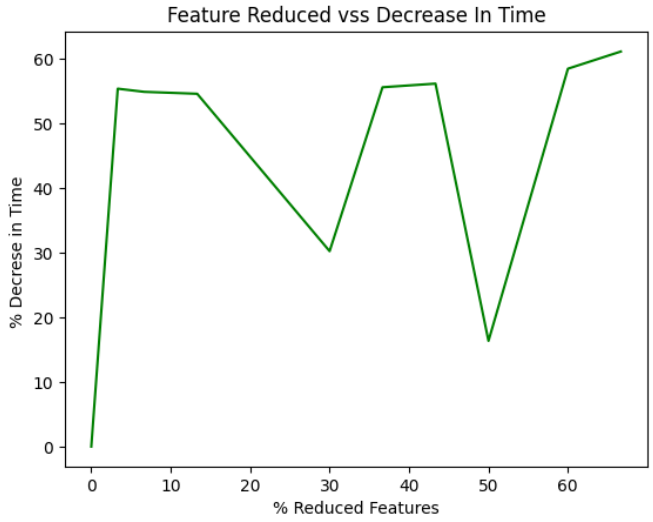

Figure 2. Plot of percentage reduction in features versus percentage reduction in time.

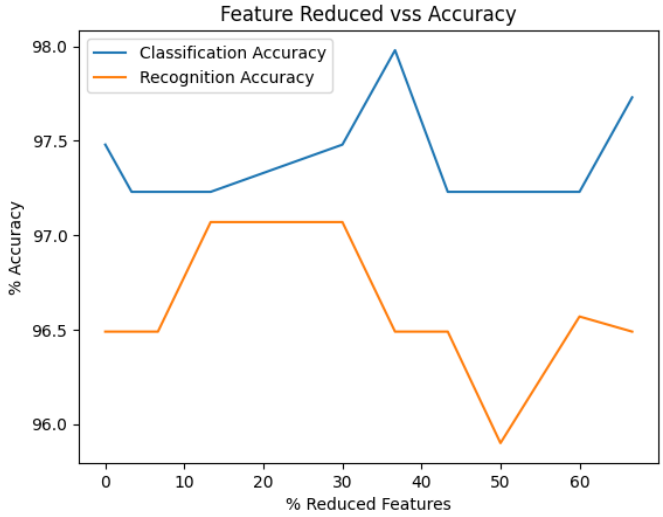

Figure 3. Effect of feature reduction on accuracy

TABLE 1. RESULTS OF THE BREAST CANCER WISCONSIN DATASET.

Total Selected Features	% Reduced Features	Classification Accuracy	Recognition Accuracy	%Reduced Time	%(Classification-Recognition)
30	-	97.48	96.49	-	1.01
29	3.33	97.23	96.49	55.40	0.76
28	6.66	97.23	96.49	54.91	0.76
26	13.33	97.23	97.07	54.61	0.16
21	**30.00**	**97.48**	**97.07**	**30.25**	**0.42**
19	36.66	97.98	96.49	55.62	1.52
17	43.33	97.23	96.49	56.18	0.76
15	50.00	97.23	95.90	16.36	1.36
12	60.00	97.23	96.57	58.49	0.67

B. Wine Quality Dataset

Wine quality dataset contains 11 independent variables and one output variable with 2 classes. Features in this dataset have negligible correlation with each other, in such cases, our method plays an important role in reducing the dimensionality. Figure 4 shows the effect of reduction in features on classification and recognition accuracy. The blue and orange line in the graph represents the effect of feature reduction on classification and recognition accuracy respectively. Table [2] shows experimental results of our KNN model including effects on accuracy and time. For this dataset we have achieved a 26.17% reduction in time when 18.18% features were deducted by maintaining accuracy of the model. For 0% reduction in features classification accuracy is 76.54%. When the reduction percentage is increased to 9.09% classification accuracy decreases to 74.74% but recognition accuracy increases. In Figure 4, for the highlighted row with 18.18% reduction optimum solution is obtained where classification accuracy is better than the original accuracy, also recognition accuracy is increased significantly and reduction in recognition time is 26.17%. Further reduction is leading to non-optimal solutions.

TABLE 2. RESULTS OF THE WINE QUALITY DATASET

Total Selected Features	% Reduced Features	Classification Accuracy	Recognition Accuracy	%Reduced Time	%(Classification-Recognition)
11	-	76.54	71.25	-	6.91
10	9.09	74.74	71.87	35.46	3.83
9	**18.18**	**75.76**	**73.75**	**26.17**	**2.65**
8	27.27	75.68	68.12	42.24	9.40
7	36.36	74.27	70.93	46.06	4.49
5	54.54	72.32	71.56	53.55	1.05
4	63.36	73.02	70.62	55.38	3.28
2	81.81	72.32	70.62	54.87	2.35

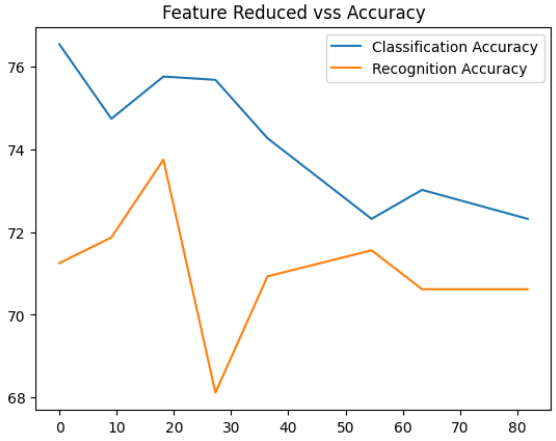

Figure 4. Effect of feature reduction on accuracy

We tested our model using the dry beans dataset [19] obtained from Kaggle which contains 16 independent variables and 13,611 data samples and output has 7 classes, in order to validate our approach. Table [3] shows that by reducing approximately 33% of features, we achieved a comparable accuracy with only a 2.45% decrease. Moreover, there is no notable decline in recognition accuracy, thus aligning with our objectives.

TABLE 3. RESULTS OF THE DRY BEANS DATASET.

Total Selected Features	% Reduced Features	Classification Accuracy	Recognition Accuracy	%Reduced Time	%(Classification-Recognition)
15	-	92.74	92.47	-	0.27
14	6.66	92.10	91.99	20.20	0.11
13	13.33	90.17	89.16	25.46	1.01
12	20.00	90.21	89.34	32.15	0.87
11	26.66	90.18	89.20	33.24	0.98
10	**33.33**	**90.29**	**89.74**	**33.46**	**0.55**
9	40.00	90.24	89.68	34.30	0.56
8	46.66	90.29	89.38	37.11	0.91
7	53.33	89.44	88.39	39.51	1.05
6	60.00	89.30	88.39	43.10	0.92

Similarly, we verified our model for one more dataset. For the diabetes dataset, we achieved nearly a 25% reduction in the number of features which reduced time by nearly 25%. Overall, we can reduce 20 to 30 percent of features from the dataset using our method by maintaining original accuracy. It can be concluded from the results; that our model can reduce nearly 25% to 30% of the recognition time. Our results show that our model successfully reduced significant recognition time by maintaining the accuracy of the model.

From the last column of Table [1] and Table [2] it can be observed that when we decrease the number of features, the recognition accuracy of the model is maintained at the optimal solution and the percentage decrease in recognition accuracy from classification accuracy also decreases at the optimum solution this indicates that we are getting better recognition accuracy when we decrease features. So, it can be concluded that when we train models with fewer features, then recognition of data labels on new data becomes easy and hence, we get better recognition accuracy.

High-dimensional datasets are becoming increasingly prevalent, capturing complex interactions between numerous factors and over 16 million users from 190 countries are using Kaggle datasets for their research work. However, processing and analyzing them presents a significant hurdle, primarily in the form of excessive computational requirements. This high computational cost translates directly to a potential carbon footprint, raising crucial questions about the environmental impact of studying global warming. Our proposed approach offers a solution by reducing the dimensionality of these datasets. This diminishes computational complexity, consequently mitigating the aforementioned environmental concerns. This, in turn, leads to a lower energy footprint, contributing to environmentally sustainable climate research by minimizing the carbon footprint associated with enterprise-level computing tasks.

V. CONCLUSION

In conclusion, this study gives an innovative approach for handling ‘curse of dimensionality’ by enhancing the efficiency of the KNN algorithm by employing feature reduction based on standard deviation. The key takeaway from this research is that intelligent feature selection, guided by standard deviation analysis, can substantially reduce the recognition time without compromising the accuracy of the predictions. Through comprehensive experiments on diverse datasets, including breast cancer, wine quality and diabetes, the results showed that a reduction of approximately 18% to 30% in features led to a reduction in average recognition time by 27%. This reduction in recognition time makes the KNN algorithm more suitable for real-time applications and large-scale datasets.

VI. Future Scope

Investigating hybrid methods that combine standard deviation-based feature reduction with other dimensionality reduction techniques, such as PCA [8] or t-Distributed Stochastic Neighbor Embedding (t-SNE) [19], might further improve the efficiency and accuracy of the KNN algorithm. Also extending this research to other machine learning algorithms and comparing the performance with KNN. The proposed method assumes that the datasets are balanced, so developing methods to handle class imbalances effectively while reducing features, ensuring that minority class samples are not disproportionately affected could be dealt with in the future.

References

[1] T. Cover and P. Hart, "Nearest neighbor pattern classification," in IEEE Transactions on Information Theory, vol. 13, no. 1, pp. 21-27, January 1967, doi: 10.1109/TIT.1967.1053964.

[2] https://www.kaggle.com

[3] https://www.kaggle.com/datasets/uciml/breast-cancer-wisconsin-data

[4] https://www.kaggle.com/datasets/nareshbhat/wine-quality-binary-classification

[5] https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database

[6] L. Chen, "Curse of dimensionality," in *Springer eBooks*, 2009, pp. 545–546. doi: 10.1007/978-0-387-39940-9_133.

[7] K. Beyer, J. Goldstein, R. Ramakrishnan, and U. Shaft, "When is 'Nearest Neighbor' meaningful?" in Lecture Notes in Computer Science, 1999, pp. 217–235. doi: 10.1007/3-540-49257-7_15.

[8] S. Wold, K. Esbensen, P. Geladi, "Principal component analysis," *Chemometr. Intell. Lab. Syst.*, vol. 2, no. 1–3, pp. 37–52, 1987.

[9] G. Borowik, T. Łuba and R. Klempous, "Comparison of algorithms for dimensionality reduction and their application to index generation functions," 2020 IEEE 15th International Conference of System of Systems Engineering (SoSE), Budapest, Hungary, 2020, pp. 283-288, doi: 10.1109/SoSE50414.2020.9130484.

[10] Y. Song et al., "A New Parameterized Algorithm for Accurate Supervised Dimensionality Reduction," 2018 8th International Conference on Electronics Information and Emergency Communication (ICEIEC), Beijing, China, 2018, pp. 166-169, doi: 10.1109/ICEIEC.2018.8473475.

[11] M. Espadoto, R. M. Martins, A. Kerren, N. S. T. Hirata and A. C. Telea, "Toward a Quantitative Survey of Dimension Reduction Techniques," in IEEE Transactions on Visualization and Computer Graphics, vol. 27, no. 3, pp. 2153-2173, 1 March 2021, doi: 10.1109/TVCG.2019.2944182.

[12] Mladenić, Dunja. "Feature selection for dimensionality reduction." International Statistical and Optimization Perspectives Workshop" Subspace, Latent Structure and Feature Selection". Berlin, Heidelberg: Springer Berlin Heidelberg, 2005.

[13] S. Sivaranjani, S. Ananya, J. Aravinth and R. Karthika, "Diabetes Prediction using Machine Learning Algorithms with Feature Selection and Dimensionality Reduction," 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2021, pp. 141-146, doi: 10.1109/ICACCS51430.2021.9441935.

[14] S. Ayesha, M. K. Hanif, and R. Talib, "Overview and comparative study of dimensionality reduction techniques for high dimensional data," Information Fusion, vol. 59, pp. 44–58, Jul. 2020, doi: 10.1016/j.inffus.2020.01.005.

[15] Jia, W., Sun, M., Lian, J. et al., "Feature dimensionality reduction: a review", in Complex Intell. Syst. 8, 2663–2693 ,21 January 2021, doi:10.1007/s40747-021-00637-x

[16] N. Gopika and A. M. Kowshalaya M.E., "Correlation Based Feature Selection Algorithm for Machine Learning," 2018 3rd International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, 2018, pp. 692-695, doi: 10.1109/CESYS.2018.8723980.

[17] S. Taneja, C. Gupta, K. Goyal and D. Gureja, "An Enhanced K-Nearest Neighbor Algorithm Using Information Gain and Clustering," 2014 Fourth International Conference on Advanced Computing & Communication Technologies, Rohtak, India, 2014, pp. 325-329, doi: 10.1109/ACCT.2014.22

[18] Youqiang Zhang, Guo Cao, Bisheng Wang, Xuesong Li, A novel ensemble method for k-nearest neighbor, Pattern Recognition, Volume 85,2019,Pages 13-25,ISSN 0031-3203.

[19] https://www.kaggle.com/code/ahmadheshamzaki/dry-beans-eda

[20] Van der Maaten, Laurens, and Geoffrey Hinton. "Visualizing data using t-SNE." Journal of machine learning research 9, no. 11 (2008).