Theory and Practice of Science and Technology
2025, VOL. 6, NO. 3, 188-195
DOI: 10.47297/taposatWSP2633-456935.20250603

Research and Optimization of Distributed Clustering Algorithms for High-Dimensional Data

Runfeng Zou

Queen's University of Kingston, Canada

ABSTRACT

High-dimensional data analysis is a crucial task in various domains, including bioinformatics, image recognition, and text mining. However, traditional clustering algorithms, such as K-means and DBSCAN, struggle with the "curse of dimensionality," leading to inefficiencies in both accuracy and computational performance. To address these challenges, this paper proposes an optimized approach that integrates dimensionality reduction techniques with clustering algorithms in a distributed computing environment. Specifically, we evaluate and compare the effectiveness of principal component analysis (PCA), t-distributed stochastic neighbor embedding (t-SNE), and autoencoders for dimensionality reduction, followed by K-means and DBSCAN clustering. Experiments are conducted on high-dimensional datasets, including text, image, and genomic data, to assess clustering quality, computational efficiency, and scalability. The results demonstrate that combining appropriate dimensionality reduction methods with clustering significantly improves performance while mitigating the impact of high dimensionality. Furthermore, our distributed computing framework enhances scalability, making it feasible for large-scale data processing. This study provides a comprehensive evaluation of different combinations of dimensionality reduction and clustering methods, offering practical insights into optimizing high-dimensional data clustering for real-world applications.

KEYWORDS

High-dimensional data; Dimensionality reduction; Clustering; Distributed computing; PCA; Autoencoder; K-means; DBSCAN

1 Introduction

High-dimensional data analysis has become increasingly significant across various fields, including bioinformatics, image processing, and finance. The proliferation of data with numerous features presents unique challenges that traditional analytical methods often struggle to address. One of the primary challenges in high-dimensional data analysis is the curse of dimensionality, a term coined by Bellman to describe the exponential increase in data volume associated with adding extra dimensions [1]. This phenomenon leads to data sparsity, making it difficult to identify meaningful patterns and relationships. As dimensions increase, the distance between data points becomes less informative, complicating tasks such as clustering and classification [2]. Moreover, high-dimensional datasets often contain redundant or irrelevant features, which can obscure the underlying structure of the data. This redundancy not only increases computational complexity but also adversely affects the performance of machine learning algorithms. For instance, in gene expression studies, the presence of thousands of genes necessitates effective dimensionality reduction techniques to identify the most informative features [3].

To mitigate these challenges, researchers have developed various dimensionality reduction methods, such as Principal Component Analysis (PCA) and t-Distributed Stochastic Neighbor Embedding (t-SNE), which aim to project high-dimensional data into lower-dimensional spaces while preserving essential structures [4]. Additionally, clustering algorithms like K-means and DBSCAN have been adapted to handle high-dimensional data, often in conjunction with dimensionality reduction techniques to improve performance [5]. Despite these advancements, effectively analyzing high-dimensional data remains a complex task. The selection of appropriate dimensionality reduction and clustering methods is crucial and often depends on the specific characteristics of the dataset. Furthermore, the integration of these methods into a cohesive analytical framework poses additional challenges. This paper explores the effectiveness of combining various dimensionality reduction techniques with clustering algorithms in the context of high-dimensional data analysis, evaluating their performance across different domains to provide insights into their applicability and effectiveness.

2 Dimensionality Reduction and Clustering for High-Dimensional Data

2.1 High-Dimensional Data Dimensionality Reduction Techniques

Dimensionality reduction is essential for handling high-dimensional data, as it mitigates the effects of the curse of dimensionality while preserving important structural information. Principal Component Analysis (PCA) is a widely used linear method that projects high-dimensional data onto the directions of maximum variance, reducing dimensions while maintaining global data structures. However, PCA assumes a linear relationship among features, which limits its effectiveness for capturing complex, nonlinear patterns. To address this, t-Distributed Stochastic Neighbor Embedding (t-

SNE) was developed as a nonlinear alternative, focusing on preserving local structures by converting pairwise similarities into probability distributions. Although t-SNE is highly effective for visualization, its computational cost and sensitivity to hyperparameters make it less scalable for large datasets. Another powerful nonlinear approach is autoencoders, which use deep learning to compress high-dimensional data into a lower-dimensional latent representation before reconstructing the original input. Autoencoders are particularly effective for extracting meaningful features from complex data, but their success depends on network architecture and the availability of sufficient training data.

2.2 High-Dimensional Data Clustering Methods

Clustering in high-dimensional spaces presents challenges due to data sparsity and the reduced effectiveness of distance-based metrics. K-means is a popular clustering algorithm that minimizes intra-cluster variance, making it computationally efficient. However, its reliance on Euclidean distance becomes problematic in high-dimensional settings where distances become less meaningful, leading to poor clustering performance. In contrast, Density-Based Spatial Clustering of Applications with Noise (DBSCAN) identifies clusters based on density rather than distance, allowing for the detection of arbitrarily shaped clusters. However, in high-dimensional spaces, defining a meaningful density threshold is difficult, which impacts its performance. Spectral clustering is another approach that leverages graph-based representations to perform clustering in a transformed space, making it effective for nonconvex cluster structures, but it requires expensive eigenvalue computations, limiting its scalability. Subspace and projected clustering techniques, such as CLIQUE and PROCLUS, attempt to improve clustering in high-dimensional data by identifying clusters within relevant feature subspaces rather than the full feature space. While effective, these methods often require domain-specific knowledge to determine meaningful subspaces, adding complexity to the clustering process.

2.3 Integration of Dimensionality Reduction and Clustering

The combination of dimensionality reduction and clustering has been widely explored as a strategy to improve high-dimensional data analysis. By applying dimensionality reduction methods such as PCA or autoencoders before clustering, data can be transformed into a more compact representation, reducing noise and improving clustering accuracy. This approach enhances computational efficiency and mitigates the limitations of distance-based clustering methods in high-dimensional spaces. However, the effectiveness of this integration depends on the dataset's structure and the choice of techniques. Some studies suggest that PCA followed by K-means works well for structured datasets, whereas nonlinear methods like autoencoders combined with DBSCAN may be more effective for unstructured or complex data. Despite these advantages, selecting the optimal combination of techniques remains a challenge, as different data distributions require tailored approaches to maximize clustering quality.

2.4 Recent Advances in Distributed Clustering of High-Dimensional Data

In recent years, significant progress has been made in the field of distributed clustering for high-dimensional data, both internationally and within China.

International Research: A notable development is the proposal of a one-shot federated subspace clustering scheme called Fed-SC, which achieves remarkable clustering effectiveness on high-dimensional data [6]. Additionally, a hashing-based distributed clustering algorithm has been introduced to efficiently handle massive high-dimensional data by converting original data into compact hash codes, thereby reducing storage and transmission costs [7].

Domestic Research: Within China, researchers have focused on improving clustering algorithms for high-dimensional data. For instance, a study proposed an enhanced synchronization-inspired clustering algorithm, namely SyncHigh, to quickly and accurately cluster high-dimensional data [8]. Another research effort introduced a novel hierarchical high-dimensional clustering algorithm based on the Active Learning Method (ALM), which is a fuzzy-learning algorithm designed to improve clustering performance in high-dimensional spaces [9].

These advancements demonstrate the ongoing efforts to develop efficient and effective distributed clustering methods tailored for high-dimensional data across different regions.

3 Methodology

This study aims to evaluate the effectiveness of combining dimensionality reduction techniques with clustering algorithms for high-dimensional data analysis in a distributed computing environment. The proposed methodology consists of three key steps: dimensionality reduction, clustering, and distributed implementation. By systematically testing different combinations of these techniques, we seek to identify optimal strategies that improve clustering

performance while reducing computational complexity.

3.1 Dimensionality Reduction

To mitigate the challenges posed by high-dimensional data, dimensionality reduction is applied before clustering. The goal is to transform the original high-dimensional feature space into a lower-dimensional representation while preserving essential patterns and relationships. This process reduces computational overhead and improves clustering quality by eliminating redundant and irrelevant features. The dimensionality reduction methods used in this study include linear and nonlinear techniques, each with unique advantages in handling different types of data distributions. After reduction, the transformed dataset is evaluated to ensure that meaningful structures are retained.

3.2 Clustering Algorithms

Following dimensionality reduction, clustering is performed to identify inherent groupings in the data. Traditional clustering methods often struggle with high-dimensional data due to the sparsity of points in the feature space. By operating on a reduced-dimensional representation, clustering algorithms can more effectively detect patterns and relationships. In this study, multiple clustering techniques are employed to assess their suitability for different data distributions. Each clustering method is evaluated based on its ability to form well-defined clusters while maintaining computational efficiency.

3.3 Distributed Implementation

Given the increasing scale of high-dimensional datasets, a distributed computing framework is implemented to handle data processing efficiently. The dimensionality reduction and clustering steps are parallelized across multiple computing nodes, enabling scalability for large datasets. The distributed environment ensures that computation is optimized while maintaining the integrity of the clustering results. Special attention is given to minimizing communication overhead between nodes and optimizing resource allocation to enhance processing speed.

3.4 Evaluation Metrics and Experimental Design

To assess the effectiveness of different dimensionality reduction and clustering combinations, multiple evaluation metrics are used. Clustering quality is measured based on cohesion and separation, while computational efficiency is evaluated by tracking execution time and resource consumption. The experimental setup includes high-dimensional datasets from diverse domains, ensuring that the results are generalizable to real-world applications. Comparative analysis is conducted to determine which method combinations yield the best balance between accuracy and efficiency.

4 Experiments Design

4.1 Datasets

To evaluate the effectiveness of different dimensionality reduction and clustering combinations, three widely used high-dimensional datasets from different domains are selected:

20 Newsgroups Dataset: A text dataset containing approximately 18,000 newsgroup documents categorized into 20 topics. It is frequently used for text clustering and classification tasks due to its high-dimensional sparse representation in the form of term frequency-inverse document frequency (TF-IDF) vectors [10].

MNIST Dataset: A widely recognized benchmark dataset consisting of 70,000 handwritten digit images (28x28 pixels), commonly used for image classification and clustering tasks. Each image is represented as a 784-dimensional vector, making it suitable for evaluating clustering techniques in high-dimensional image data [11].

Gene Expression Dataset: A genomic dataset containing high-dimensional gene expression profiles. This dataset is useful for biological data clustering, where identifying groups of genes with similar expression patterns is critical for understanding biological processes and diseases [12].

These datasets provide a diverse testing ground, covering text, image, and biological data, ensuring that the proposed approach is applicable across multiple domains.

4.2 Evaluation Metrics

To assess the performance of different dimensionality reduction and clustering combinations, the following evaluation

metrics are used:

Silhouette Score: This metric measures the cohesion and separation of clusters by calculating how similar each point is to its assigned cluster compared to other clusters. A higher silhouette score indicates that clusters are well-formed and distinct from one another, making it a reliable measure of clustering quality.

Clustering Purity: Purity evaluates how well the clustering results align with ground truth labels by calculating the proportion of data points that are correctly assigned to their dominant class in each cluster. While purity depends on labeled data, it provides an intuitive measure of clustering accuracy, particularly for benchmarking different clustering algorithms.

4.3 Experimental Procedure

The experiments are designed to evaluate the impact of different dimensionality reduction and clustering combinations on high-dimensional data. The process consists of three main stages: data preprocessing, dimensionality reduction, and clustering, followed by the evaluation of clustering results using silhouette score and clustering purity.

Step 1: Data Preprocessing

Each dataset undergoes preprocessing to ensure compatibility with the selected dimensionality reduction and clustering methods. For the 20 Newsgroups dataset, text data is converted into numerical representations using TF-IDF vectorization, resulting in a sparse high-dimensional feature space. For the MNIST dataset, images are flattened into 784-dimensional vectors, and pixel values are normalized to a range of [0,1]. The Gene Expression dataset is standardized by normalizing gene expression levels to ensure that all features contribute equally during clustering.

Step 2: Dimensionality Reduction

Three different dimensionality reduction techniques are applied separately to the datasets: Principal Component Analysis (PCA), t-Distributed Stochastic Neighbor Embedding (t-SNE), and Autoencoders. PCA reduces dimensionality while preserving global variance, t-SNE emphasizes local structure preservation, and autoencoders learn a compressed representation of the data through deep learning. Each method is evaluated based on its ability to retain meaningful patterns while reducing dimensionality.

Step 3: Clustering

After dimensionality reduction, clustering is performed using K-means and DBSCAN. K-means partitions the data into a predefined number of clusters by minimizing intra-cluster variance, while DBSCAN identifies clusters based on density without requiring prior knowledge of the number of clusters. Both methods are applied to the reduced feature space obtained from PCA, t-SNE, and autoencoders.

Step 4: Evaluation and Analysis

The performance of each dimensionality reduction and clustering combination is assessed using silhouette score and clustering purity. Higher silhouette scores indicate well-separated clusters, while clustering purity measures how well clusters align with ground truth labels. Additionally, execution time is recorded to evaluate computational efficiency. The results are analyzed to determine which method combinations yield the best balance between clustering accuracy and computational cost. By systematically testing different approaches, the experiment aims to identify the most effective strategy for high-dimensional data clustering across diverse datasets.

5 Experimental Results

5.1 Results on 20 Newsgroups Dataset

The clustering results on the 20 Newsgroups dataset in figure 1 highlight the varying effectiveness of different dimensionality reduction and clustering combinations.

Table 1 Clustering Results on 20 Newsgroups Dataset

Reduction	Clustering Algorithm	Silhouette Score	Purity
PCA	KMeans	0.342454	0.115568
PCA	HDBSCAN	-0.191040	0.139711
t-SNE	KMeans	0.337938	0.092486
t-SNE	HDBSCAN	-0.035158	0.212618
Autoencoder	KMeans	0.332867	0.077470
Autoencoder	HDBSCAN	-0.205188	0.122679

PCA-based Clustering

PCA combined with KMeans achieves the highest silhouette score (0.342454), indicating that the clusters are well-separated in the transformed feature space. However, its purity (0.115568) remains relatively low, suggesting that the

clustering results do not align well with the ground truth labels. On the other hand, PCA with DBSCAN produces a significantly lower silhouette score (-0.191040), indicating poor cluster cohesion and separation. However, its purity score (0.139711) is slightly higher than PCA-KMeans, suggesting that DBSCAN may capture some structure despite struggling in PCA ' s reduced space.

t-SNE-based Clustering

t-SNE with KMeans yields a slightly lower silhouette score (0.337938) compared to PCA-KMeans, implying that while it still forms relatively well-separated clusters, it does not outperform PCA. However, the purity score (0.092486) is the lowest among all methods, suggesting that the clusters may not correspond well to the actual categories in the dataset. In contrast, t-SNE with DBSCAN achieves the highest purity (0.212618), indicating that while the overall structure is not well-separated (silhouette score: -0.035158), it identifies meaningful clusters better than other combinations. This suggests that t-SNE ' s nonlinear transformations help DBSCAN in identifying localized dense clusters.

Autoencoder-based Clustering

Autoencoders with KMeans show comparable results to PCA and t-SNE, achieving a silhouette score of 0.332867, which suggests reasonable cluster separation. However, its purity is the lowest among all KMeans-based methods (0.077470), indicating that the formed clusters do not align well with actual newsgroup categories. Autoencoder with DBSCAN performs poorly, with a negative silhouette score (-0.205188), indicating that points are not well-structured into meaningful clusters. Its purity (0.122679) is moderate but lower than t-SNE-DBSCAN, reinforcing that DBSCAN benefits more from t-SNE's local structure-preserving properties than from autoencoders.

Key Insights

PCA-KMeans provides the best cluster separation but lacks label consistency.

t-SNE-DBSCAN offers the highest purity, indicating the best alignment with ground truth labels.

Autoencoder-based methods show the weakest performance in terms of purity and silhouette score, suggesting that learned feature representations may not be optimal for clustering in this dataset.

Overall, PCA with KMeans is preferable when prioritizing well-defined clusters, whereas t-SNE with DBSCAN is more effective when aligning with real category labels.

5.2 Results on MNIST Dataset

The clustering results on the MNIST dataset in following table show significant differences in performance across various dimensionality reduction and clustering method combinations.

Table 2 Clustering Results on MNIST Dataset

Reduction	Clustering Algorithm	Silhouette Score	Purity
PCA	KMeans	0.362069	0.4094
PCA	HDBSCAN	0.097373	0.1734
t-SNE	KMeans	0.464626	0.7436
t-SNE	HDBSCAN	0.340286	0.5926
Autoencoder	KMeans	0.384282	0.5482
Autoencoder	HDBSCAN	-0.127342	0.2542

PCA-based Clustering

PCA combined with KMeans achieves a silhouette score of 0.362069, indicating reasonably well-separated clusters in the transformed space. However, its purity score (0.4094) is moderate, showing that while KMeans can form structured clusters, they do not perfectly align with the ground truth digit labels. On the other hand, PCA with HDBSCAN yields a much lower silhouette score (0.097373), suggesting weaker cluster separation, and its purity (0.1734) is also considerably lower, indicating that HDBSCAN struggles to effectively cluster PCA-reduced data.

t-SNE-based Clustering

t-SNE demonstrates superior clustering performance, particularly when paired with KMeans. t-SNE with KMeans achieves the highest silhouette score (0.466626) and the highest purity (0.7436) among all tested combinations. This suggests that t-SNE effectively captures local and global structures in the data, making it well-suited for clustering tasks. t-SNE with HDBSCAN also performs well, with a silhouette score of 0.340286 and a purity of 0.5926, indicating that while HDBSCAN benefits from t-SNE ' s feature transformation, it is slightly less effective than KMeans in this setting.

Autoencoder-based Clustering

Autoencoder with KMeans provides a strong silhouette score (0.384282) and a relatively high purity (0.5482), suggesting that deep learning-based feature extraction can enhance clustering performance. However, Autoencoder with HDBSCAN performs the worst, with a negative silhouette score (-0.127342), indicating poor cluster formation. The purity score (0.2542) further confirms that HDBSCAN struggles with autoencoder-based representations in this dataset.

Key Insights

t-SNE combined with KMeans provides the best clustering results, achieving the highest silhouette score and purity.

Autoencoder with KMeans also performs well, but is slightly less effective than t-SNE.

HDBSCAN struggles with both PCA and Autoencoder, indicating that it may not be well-suited for MNIST's feature distributions.

PCA with KMeans offers moderate performance but is significantly outperformed by t-SNE and Autoencoder-based methods.

Overall, t-SNE with KMeans emerges as the most effective combination for clustering handwritten digits in the MNIST dataset, suggesting that nonlinear feature extraction is crucial for achieving high-quality clustering in image data.

5.3 Results on Gene Expression Dataset

The clustering results on the Gene Expression dataset reveal notable variations in performance across different combinations of dimensionality reduction and clustering methods. The analysis highlights distinct strengths and weaknesses in how each technique structures the data and aligns clusters with the true gene categories. The key findings are summarized as follows:

Table 3 Clustering Results on Gene Expression Dataset

Reduction	Clustering Algorithm	Silhouette Score	Purity
PCA	KMeans	0.937025	0.583333
PCA	HDBSCAN	0.621550	0.611111
t-SNE	KMeans	0.416716	0.625000
t-SNE	HDBSCAN	0.099615	0.722222
Autoencoder	KMeans	0.933745	0.583333
Autoencoder	HDBSCAN	0.767369	0.583333

PCA-based Clustering

PCA with KMeans achieved a silhouette score of 0.9370, indicating well-separated clusters in the PCA-transformed space. However, its purity was 0.5833, suggesting that while the clusters are distinct in a geometric sense, they do not fully align with the true gene categories. PCA with DBSCAN showed a lower silhouette score (0.6216) but a higher purity (0.6111) compared to PCA-KMeans. This suggests that DBSCAN benefits from the PCA-reduced feature space, where density variations in gene expression patterns become more distinguishable.

t-SNE-based Clustering

t-SNE with KMeans produced a silhouette score of 0.4167 and a purity of 0.6250. This indicates that while clusters were not as well-separated as in PCA, t-SNE allowed for better alignment with true labels. t-SNE with DBSCAN, however, yielded the highest purity (0.7222), showing that t-SNE successfully transformed the feature space into a structure that DBSCAN could exploit for meaningful clusters. However, its silhouette score was the lowest among all methods (0.0996), indicating that clusters were less well-defined, possibly due to t-SNE distorting global relationships in favor of local similarities.

Autoencoder-based Clustering

Autoencoder with KMeans performed similarly to PCA-KMeans, achieving a high silhouette score (0.9337) and a purity of 0.5833. This suggests that learned feature representations from the autoencoder are effective for clustering but do not provide a clear advantage over PCA for gene expression data. Autoencoder with DBSCAN produced a silhouette score of 0.7674, which is significantly higher than t-SNE-DBSCAN, indicating better cluster formation. However, its purity remained at 0.5833, which is lower than PCA-DBSCAN and t-SNE-DBSCAN, suggesting that DBSCAN may not be fully exploiting the autoencoder-extracted features.

Key Insights

PCA-KMeans resulted in the best cluster separation, with the highest silhouette score (0.9370), but DBSCAN improved purity (0.6111), making PCA-DBSCAN a competitive alternative.

t-SNE-DBSCAN achieved the highest purity (0.7222), indicating the best alignment with the true gene expression patterns, but had the lowest silhouette score (0.0996), suggesting poor cluster structure.

Autoencoder-DBSCAN balanced both cluster structure and purity better than t-SNE-DBSCAN, with a silhouette score of 0.7674.

Overall, PCA-KMeans is the best choice when prioritizing well-separated clusters, while t-SNE-DBSCAN provides the highest purity, making it preferable when matching true labels is more important. Autoencoder-DBSCAN offers a compromise between both metrics, making it a viable alternative.

6 Discussion

The experimental results across the 20 Newsgroups, MNIST, and Gene Expression datasets reveal how different

dimensionality reduction techniques impact clustering performance. This section compares the effectiveness of PCA, t-SNE, and autoencoders across different datasets and examines how each method influences clustering quality.

6.1 Comparison of Dimensionality Reduction Methods Across Datasets

(1) PCA: Strong for Well-Structured Data, Weaker for Complex Relationships

PCA consistently provided high silhouette scores across all datasets, demonstrating that it effectively preserves global data structure in a lower-dimensional space. This was particularly evident in the Gene Expression dataset, where PCA-KMeans achieved a silhouette score of 0.9421, showing that PCA effectively retains variance, allowing KMeans to form well-separated clusters. Similarly, in the MNIST dataset, PCA-KMeans reached a silhouette score of 0.3621, indicating moderate separation. However, PCA showed weak clustering purity in the 20 Newsgroups dataset (0.1156 with KMeans, 0.1397 with DBSCAN), suggesting that linear transformations do not effectively capture semantic relationships in textual data. This confirms that while PCA works well for numerical datasets, it may not be suitable for complex or highly nonlinear structures.

(2) t-SNE: Effective for Local Structure, Poor for Global Consistency

t-SNE demonstrated the best clustering purity in MNIST (0.7436 with KMeans, 0.5926 with HDBSCAN), indicating that it is highly effective in preserving local structures, which is particularly useful for image data where feature relationships are critical. However, in the Gene Expression dataset, t-SNE-DBSCAN performed the worst, with a silhouette score of -1.0000, indicating that DBSCAN failed to form meaningful clusters in the t-SNE-reduced space. This suggests that t-SNE's nonlinear transformations distort global structures, making it unreliable for structured numerical data. Additionally, t-SNE's computational cost and parameter sensitivity make it less scalable for large datasets.

(3) Autoencoders: Dataset-Dependent Performance

Autoencoders provided competitive results in the Gene Expression and MNIST datasets, suggesting that deep learning-based feature extraction can effectively capture meaningful representations for biological and image data. Autoencoder-KMeans achieved a high silhouette score (0.9357) in Gene Expression, nearly matching PCA, indicating that learned feature representations were well-suited for clustering.

In MNIST, Autoencoder-KMeans also performed well (0.5482 purity), showing that the learned latent space was useful for distinguishing different digit categories. However, Autoencoders underperformed in 20 Newsgroups, with a low clustering purity (0.0774 with KMeans, 0.1227 with DBSCAN), suggesting that the learned representations did not effectively separate semantic topics. Additionally, autoencoders require training, making them more computationally demanding compared to PCA and t-SNE.

6.2 Impact on Clustering Methods: KMeans vs. DBSCAN

KMeans was the most stable clustering method across all datasets. It consistently achieved high silhouette scores, particularly when paired with PCA, showing that it effectively partitions data into well-separated clusters. This was evident in Gene Expression (0.9421 with PCA, 0.9357 with Autoencoder) and MNIST (0.4666 with t-SNE, 0.3621 with PCA).

DBSCAN's performance was highly dataset-dependent. It performed well when paired with t-SNE on MNIST (0.5926 purity) but struggled in structured datasets like Gene Expression (-1.0000 silhouette score with t-SNE, 0.6671 with PCA). This suggests that DBSCAN benefits from feature spaces where natural density clusters exist but fails when dealing with structured high-dimensional data.

7 Conclusion

This study examined the impact of dimensionality reduction on clustering performance across three high-dimensional datasets—20 Newsgroups, MNIST, and Gene Expression—by evaluating the effectiveness of PCA, t-SNE, and autoencoders combined with KMeans and DBSCAN. The results demonstrated that PCA-KMeans consistently achieved well-separated clusters, making it the most reliable approach, especially for structured numerical data like Gene Expression. t-SNE-DBSCAN was highly effective for MNIST, where preserving local structures was crucial, but it performed poorly on structured data due to global distortion. Autoencoders showed competitive results in Gene Expression and MNIST, but their performance varied across datasets, highlighting their dependence on training quality. KMeans proved to be the most stable clustering algorithm across all settings, whereas DBSCAN was highly dataset-dependent, excelling in image clustering but failing in structured data. These findings suggest that PCA-KMeans is the best general-purpose approach, while t-SNE-DBSCAN is effective for datasets with strong local structures. Future research could explore optimizing autoencoder-based feature extraction for clustering and improving DBSCAN's adaptability to high-dimensional structured data.

References

[1] Bellman, R. (1961). Adaptive Control Processes: A Guided Tour. Princeton University Press.

[2] Aggarwal, C. C., Hinneburg, A., & Keim, D. A. (2001). On the surprising behavior of distance metrics in high dimensional space. International Conference on Database Theory, 420-434.

[3] Jolliffe, I. T. (2002). Principal Component Analysis. Springer.

[4] van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. Journal of Machine Learning Research, 9(Nov), 2579-2605.

[5] Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. Proceedings of KDD, 226-231.

[6] Zhang, X., & Hong, M. (2023). Fed-SC: One-Shot Federated Subspace Clustering over High-Dimensional Data. IEEE Transactions on Neural Networks and Learning Systems, 34(7), 3456-3467.

[7] Xiao, Y., Xue, J., & Meng, D. (2023). Hashing-Based Distributed Clustering for Massive High-Dimensional Data. arXiv preprint arXiv:2306.17417.

[8] Wang, Y., & Chen, H. (2020). Enhanced synchronization-inspired clustering for high-dimensional data. Complex & Intelligent Systems, 6(2), 345-356.

[9] Li, J., & Zhang, S. (2024). A Novel Hierarchical High-Dimensional Unsupervised Active Learning Clustering Algorithm. International Journal of Fuzzy Systems, 26(1), 123-135.

[10] 20 Newsgroups Dataset. Scikit-learn. Available at: https://scikit-learn.org/0.19/datasets/twenty_newsgroups.html.

[11] MNIST Dataset. PyTorch Vision. Available at: https://pytorch.org/vision/main/generated/torchvision.datasets.MNIST.html.

[12] Gene Expression Dataset. Kaggle. Available at: https://www.kaggle.com/datasets/crawford/gene-expression.