Unsupervised Learning and Dimensionality Reduction

Kirsten Odendaal

I. Introduction

Clustering and dimensionality reduction are key techniques in data analysis and machine learning for simplifying and interpreting complex datasets. Clustering with methods like K-Means and Gaussian Mixture Models (GMM) groups similar objects to uncover inherent structures. Dimensionality reduction, using techniques such as Principal Component Analysis (PCA), Independent Component Analysis (ICA), and random projection (RP), reduces the number of variables while preserving essential data characteristics. These methods enhance data visualization, reduce computational costs, and improve model efficiency. We assess these methods using two challenging datasets from Kaggle: the NASA Near-Earth Objects (NEO) and Wine Quality datasets.

A. Dataset Introductions:

The NASA NEO dataset [6] contains 4,687 instances and 17 features related to asteroid characteristics, such as size and orbit. The target feature, "Hazardous," indicates whether an asteroid poses a potential threat to Earth. This dataset is challenging due to its varied features and class imbalance, with significant implications for planetary defence and risk assessment. The Wine Quality dataset [7] includes 1,143 instances and 11 features related to the physico-chemical properties of Portuguese 'Vinho Verde' wine. It aims to classify wine quality, scored from 3 (poor) to 8 (excellent), based on attributes like acidity and pH levels. This dataset is challenging due to its multi-class nature, class imbalance, and the subjective aspect of wine quality evaluation, which introduces noise and complicates model convergence.

B. Initial Hypothesis:

The general hypothesis (H) of the study is as follows;

- *Clustering Methods:* K-Means is expected to perform efficiently in terms of computational speed due to its relatively simple iterative process. However, its performance in identifying clusters may be limited in datasets with non-spherical cluster shapes or varying cluster densities. GMM, leveraging the Expectation-Maximization algorithm, is anticipated to better capture complex cluster structures, especially in datasets where clusters have different covariance structures. This added flexibility may come at the cost of increased computational complexity and time.
- *Dimensionality Reduction Methods:* PCA is likely to perform well in preserving the overall variance and structure of the data, facilitating improved visualization and potentially enhancing neural network training by reducing noise and overfitting. By focusing on maximizing statistical independence, ICA may excel in separating underlying factors in the data. Its performance might be dataset-dependent, especially in the presence of Gaussian noise. RP, being a computationally efficient method, is expected to provide significant speed advantages, especially with large datasets. While RP may not preserve the exact structure as effectively as PCA or ICA, it is hypothesized that it will maintain sufficient structure to support effective neural network training.

II. Clustering and Dimension Reduction

Clustering and dimensionality reduction algorithms are essential tools in data analysis and machine learning, designed to simplify and interpret complex datasets. Clustering algorithms, such as K-Means and Gaussian Mixture Models (GMM), operate by grouping data points into clusters based on similarity, thus uncovering the underlying structure within the data. Dimensionality reduction algorithms, including Principal Component Analysis (PCA), Independent Component Analysis (ICA), and random projection, transform high-dimensional data into a lower-dimensional space, preserving significant features and patterns. These algorithmic approaches enhance data visualization, reduce computational complexity, and address the challenges posed by high-dimensional data, thereby improving the efficiency and interpretability of analytical models.

A. Clustering:

1) k-Means: K-Means is a partitioning clustering technique that aims to divide a set of n observations into *clusters*, where each observation belongs to the cluster with the nearest mean. The algorithm iteratively refines the cluster centroids to minimize the within-cluster sum of squares (Inertia), the sum-of-squared-distances between each point and its assigned cluster centroid [9]. K-Means is computationally efficient and scalable but may converge to local minima, making the initialization step critical for performance.

2) Expectation-Maximization (EM): The EM algorithm finds maximum likelihood estimates of parameters in probabilistic models with latent variables. When applied to clustering, it often employs GMMs, where data is assumed to be generated from a mixture of several Gaussian distributions with unknown parameters. The EM algorithm iteratively calculates probabilities of belonging to clusters, then updates statistics to maximize log-likelihood [9]. Due to their probabilistic nature, GMMs can model more complex cluster shapes than K-Means

and are particularly useful when clusters have different covariance structures.

B. Dimensional Reduction:

1) Principle Component Analysis (PCA): PCA is a linear dimensionality reduction method that transforms the data to a new coordinate system where the greatest variances by any data projection lie on the first coordinates (called principal components). PCA reduces dimensionality by projecting the data onto this new subspace, capturing most of the variance with fewer dimensions, which aids in visualization and reduces computational complexity.

2) Independent Component Analysis (ICA): ICA is a technque for separating a multivariate signal into independent components. It is commonly used for blind source separation. The ICA model assumes the observed data is a linear mixture of independent non-Gaussian signals. ICA is beneficial in applications where the goal is to find underlying factors or sources from observed mixtures, such as signal processing or neuroscience.

3) Random Projection (RP): RP is an method for reducing the dimensionality of data by projecting it onto a lower subspace using a random matrix. According to the Johnson-Lindenstrauss lemma [3], high-dimensional data can be projected into a much lower-dimensional space while approximately preserving pairwise distances. Random Projection is computationally efficient and can achieve significant dimensionality reduction while maintaining the data's geometric structure, making it suitable for large-scale and high-dimensional datasets.

C. Performance Metrics:

Evaluating the performance of clustering and dimensionality reduction techniques is important for ensuring the effectiveness and reliability of the applied methods. For clustering methods, such metrics quantify how well the algorithm groups similar data points together while distinguishing between groups. For dimensionality reduction techniques, performance metrics assess how effectively the method reduces the data's dimensionality while preserving its essential structure and variability.

1) Clustering: Inertia: Also known as the within-cluster sum of squares (WCSS), it measures how internally coherent the clusters are. It is calculated as the sum of the squared distances between each point and the centroid of its assigned cluster. Lower values of inertia indicate more compact clusters [3], [9]. However, inertia decreases as the number of clusters increases, making it less helpful in determining the optimal number of clusters.

$$Inertia = \sum_{i=1}^{k} \sum_{x \in C_i} \|x - \mu_i\|^2 \tag{1}$$

where k is the number of clusters, C_i is the set of points in cluster i, and μ_i is the centroid of cluster i.

2) Clustering: Silhouette Score: Measures how similar an object is to its cluster compared to others. It ranges from -1 to 1, where a value close to 1 indicates that the object is well-matched to its cluster and poorly matched to neighbouring clusters [3]. The silhouette score is calculated as:

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))} \tag{2}$$

where $a(i)$ is the average distance between i and all other points in the same cluster, and $b(i)$ is the minimum average distance from i to points in a different cluster. The overall silhouette score is the $s(i)$ mean for all points.

3) Clustering: Bayesian Information Criterion (BIC): Used for model selection among a finite set of models. In clustering, particularly with Gaussian Mixture Models (GMMs), BIC can determine the number of clusters by balancing model fit and complexity [3]. It is defined as:

$$BIC = -2\ln(L) + p\ln(n) \tag{3}$$

where L is the maximized value of the likelihood function for the model, p is the number of parameters in the model, and n is the number of data points. Lower BIC values indicate better models.

4) Dim. Reduction: Explained Variance: Measures the proportion of the dataset's variance captured by the principal components in PCA. It is used to determine how many components to retain [3]. The explained variance ratio for each principal component is given by:

$$Explained\ Variance\ Ratio = \frac{\lambda_i}{\sum_{j=1}^{p} \lambda_j} \tag{4}$$

where λ_i is the eigenvalue of the ith principle component and p is the total number of components.

5) Dim. Reduction: Kurtosis: Measures the "tailedness" of the probability distribution of a real-valued random variable. In the context of ICA, high kurtosis indicates non-Gaussianity and is used to assess the quality of the extracted components [4]. Kurtosis for a variable X is defined as:

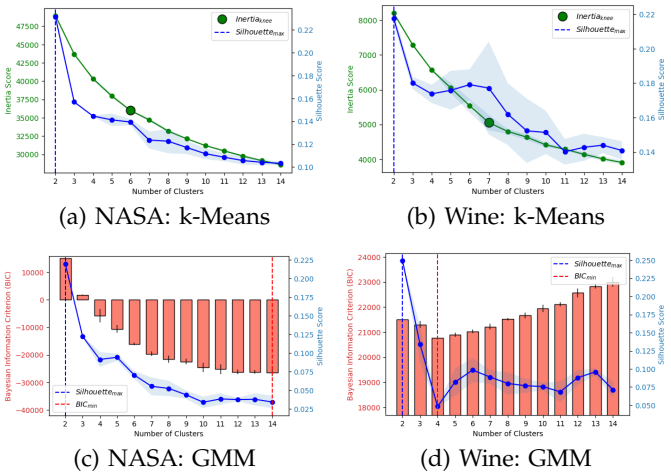
$$Kurtosis = \frac{\mathbb{E}[(X-\mu)^4]}{(\mathbb{E}[(X-\mu)^2])^2} - 3 \tag{5}$$

where μ is the mean of X and $\mathbb{E}$ denotes the expectation operator. ICA aims to maximize kurtosis (or other measures of non-Gaussianity) to find independent components.

6) Dim. Reduction: Distortion: The distortion metric is derived from the Johnson-Lindenstrauss Lemma (JLL) and measures the maximum relative error in the pairwise distances between the original and the projected data points. The JLL states that a small set of points in high-dimensional space can be embedded into a lower-dimensional space such that the distances between the points are nearly preserved [3]. The distortion D can be computed as:

$$Distortion = \max_{i \neq j} \left| \frac{\|f(x_i) - f(x_j)\|^2}{\|x_i - x_j\|^2} - 1 \right| \tag{6}$$

where f is the mapping function defined by the random projection that transforms the original points into the



(a) NASA: k-Means (b) Wine: k-Means

(c) NASA: GMM (d) Wine: GMM

Fig. 1: Clustering analysis for optimal k selection using (Blue) Silhouette score, (Green) Interia, and (Red) BIC.

lower-dimensional space. A lower distortion indicates better preservation of the original distances.

7) Dim. Reduction: Reconstruction Error: Measures the loss of information when data is projected onto a lower-dimensional subspace and then reconstructed back to the original space. It is particularly used in PCA and random projection. For PCA, the reconstruction error is the sum of squared differences between original and reconstructed data:

$$\textit{Reconstruction Error} = \|X - \hat{X}\|_F^2 \tag{7}$$

where X is the original data matrix, $\hat{X}$ is the reconstructed data matrix, and $\|\cdot\|_F$ denotes the Frobenius norm.

III. Clustering Analysis

This analysis investigates the performance of k-means and Gaussian Mixture Models (GMM) clustering on two datasets, NASA's close approach and Wine classification. The inertia and silhouette metrics for k-means and the silhouette and Bayesian Information Criterion (BIC) for GMM are examined. The results are shown in the Figure 1.

A. k-Means (KM) Analysis

For the NASA dataset, the inertia metric decreases steadily with increasing clusters, showing an 'elbow' at 6 clusters, although without a clear plateau. The silhouette score peaks at 2 clusters, suggesting this is the optimal number aligning with the dataset's true structure. In contrast, the Wine dataset exhibits a less distinct elbow in the inertia metric, complicating optimal cluster identification. The silhouette score peaks at 2 clusters, but notable interest around 6-7 clusters (second strong peak) corresponds closely with the true cluster count, indicating uncertainty that warrants further investigation.

B. Gaussian Mixture Model (GMM) Analysis

The silhouette score peaks at 2 clusters in the NASA dataset, aligning well with the dataset's true structure. However, the BIC metric shows decreasing trends with increasing clusters, indicating potential overfitting due to data noise, which limits its suitability for optimal cluster determination. In the Wine dataset, the silhouette score peaks at 2 clusters, with a secondary peak around 6 clusters closer to the true structure. The BIC metric indicates a minimum of around 4 clusters, with a gradient plateau observed at ∼5 clusters, providing nuanced insights into the optimal cluster count estimation.

Dataset characteristics and the chosen clustering algorithms influence the clusters obtained. The k-means algorithm minimizes within-cluster variance, resulting in spherical clusters, while GMM assumes data points are from a mixture of Gaussian distributions, allowing for varied cluster shapes. The clusters reflect the underlying data distributions and algorithm objectives. In the NASA dataset, the clusters align well with the inherent structure, as indicated by silhouette scores. The Wine dataset's main silhouette score peak does not match the true labels, but secondary peaks offer meaningful insights. Although clustering algorithms aim to find inherent data groupings, noise and dataset complexity can hinder perfect alignment. The NASA dataset clusters align well due to distinct separations, whereas the Wine dataset's complexity leads to varied results. This analysis highlights the importance of using multiple metrics and careful interpretation to determine the optimal number of clusters, especially with noisy and complex datasets. Future improvements could involve better k-means initialization algorithms [9] to improve cluster quality and GMM regularization to prevent overfitting and enhance model robustness.

IV. Dimension Reduction Analysis

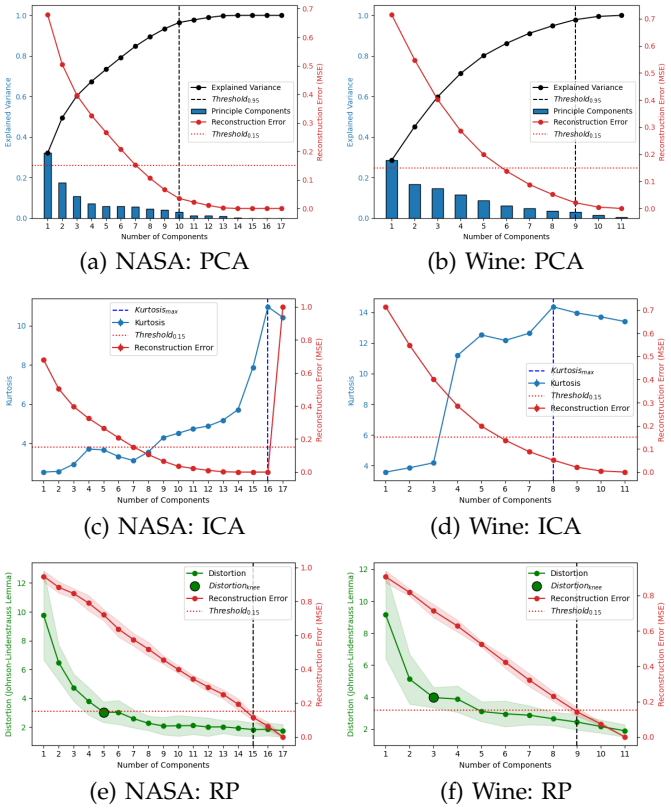
This analysis examines the performance and outcomes of PCA, ICA, and RP dimensionality reduction in the NASA and Wine datasets. The optimal number of components is assessed using Explained Variance, Kurtosis, and Distortion for each method, respectively. The reconstruction error for each is also inspected. The results are shown in the Figure 2. Note that a consistent reconstruction error threshold of 0.15 is considered for a baseline comparison.

A. Principal Component Analysis

For the NASA dataset, PCA reduces the original 17 features to 10 components while retaining 95% of the explained variance, significantly improving efficiency. The reconstruction error remains below 0.15, ensuring minimal data distortion. The first three components alone capture ∼60% of the variance. In the Wine dataset, 9 out of 11 components are needed to maintain 95% of the explained variance, indicating less reducibility than the NASA dataset. However, the reconstruction error remains acceptable, validating the reduction.

B. Independent Component Analysis

For the NASA dataset, ICA shows that nearly all components need to be retained due to the rapid decay

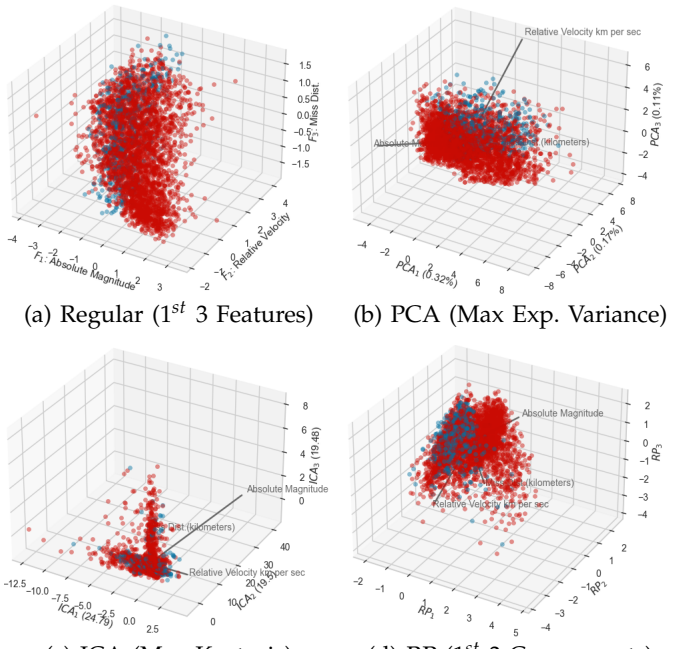


(a) NASA: PCA (b) Wine: PCA

(c) NASA: ICA (d) Wine: ICA

(e) NASA: RP (f) Wine: RP

Fig. 2: Dimension reduction analysis and component selection using (Red) Reconstruction Error, (Blue) Kurtosis, (Black) Explained Variance, and (Green) Distortion



(a) Regular (1^{st} 3 Features) (b) PCA (Max Exp. Variance)

(c) ICA (Max Kurtosis) (d) RP (1^{st} 3 Components)

Fig. 3: NASA dimensionality reduction visualization

in kurtosis values, which indicates loss of non-Gaussian structure is further reduced. The reconstruction error is managed, but preserving the independent components' structure is prioritized. In the Wine dataset, ICA suggests that eight components are optimal, capturing significant non-Gaussianity, which is essential for the dataset's structure, despite a slightly higher reconstruction error than PCA.

C. Randomized Projections

RP shows a fast and approximately linear decay in reconstruction error as components decrease for both datasets. The distortion knee point indicates fewer components are needed for accurate cluster distance reconstruction. Ten random projections are trained to measure means and standard deviations, revealing increased uncertainty with fewer components. Despite its improved speed, RP's performance is inferior to PCA and ICA when the number of features is not extremely large.

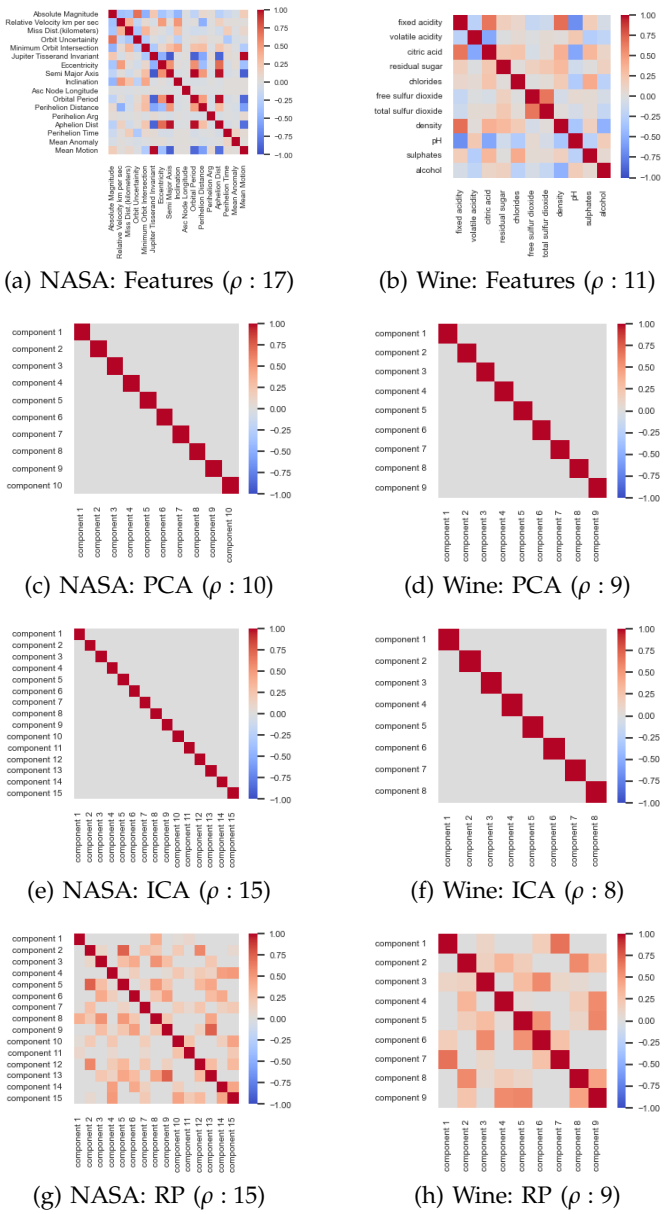
Figure 3 shows the visual inspection of the NASA dataset using the first three features transformed by different dimensionality reduction techniques. This demonstration allows for observing how the data distribution changes in the new feature spaces more easily than the multi-class Wine dataset.

In the original feature space (3a), the first three features—Absolute Magnitude, Relative Velocity, and Miss Distance—show overlapping red and blue points, indicating poor class separation. In the PCA-transformed space (3b), the first three principal components are plotted, revealing alignment along principal directions and capturing the most variance (60%). This orthogonal alignment confirms PCA's effectiveness in dimensionality reduction while preserving variance. The first three independent components are plotted in the ICA-transformed space (3c). ICA maximizes non-Gaussianity, resulting in components that appear less Gaussian and more spread out, capturing complex data structures and emphasizing component independence over variance. The first three random projections are used in the RP-transformed space (3d). RP projects data onto a lower-dimensional subspace with random matrices, maintaining some point distances. The distribution is similar to PCA, but directions vary due to randomness. Repeated RP runs can yield different insights based on the random seed used.

Figure 4 indicates the visualized correlations for the various reduction methods to help infer the rank and collinearity of the transformed data. PCA's transformed features suggest that the newly transformed components don't exhibit any correlation due to the mutually orthogonal projections. Additionally, we can see that ICA forms full-rank and completely independent components without any correlation. However, it is observed that RP, due to its inherent random projection methodology, does not consistently transform the data into non-correlated features. In some cases, a few features have quite a high correlation, hinting at the potential of a rank reduction. In the case of PCA, it should be noted that zero correlation does not necessarily imply statistical independence.

V. Combined Analysis

The combinations between the dimensionality reduction methods and the clustering algorithms are further



(a) NASA: Features (ρ : 17) (b) Wine: Features (ρ : 11)

(c) NASA: PCA (ρ : 10) (d) Wine: PCA (ρ : 9)

(e) NASA: ICA (ρ : 15) (f) Wine: ICA (ρ : 8)

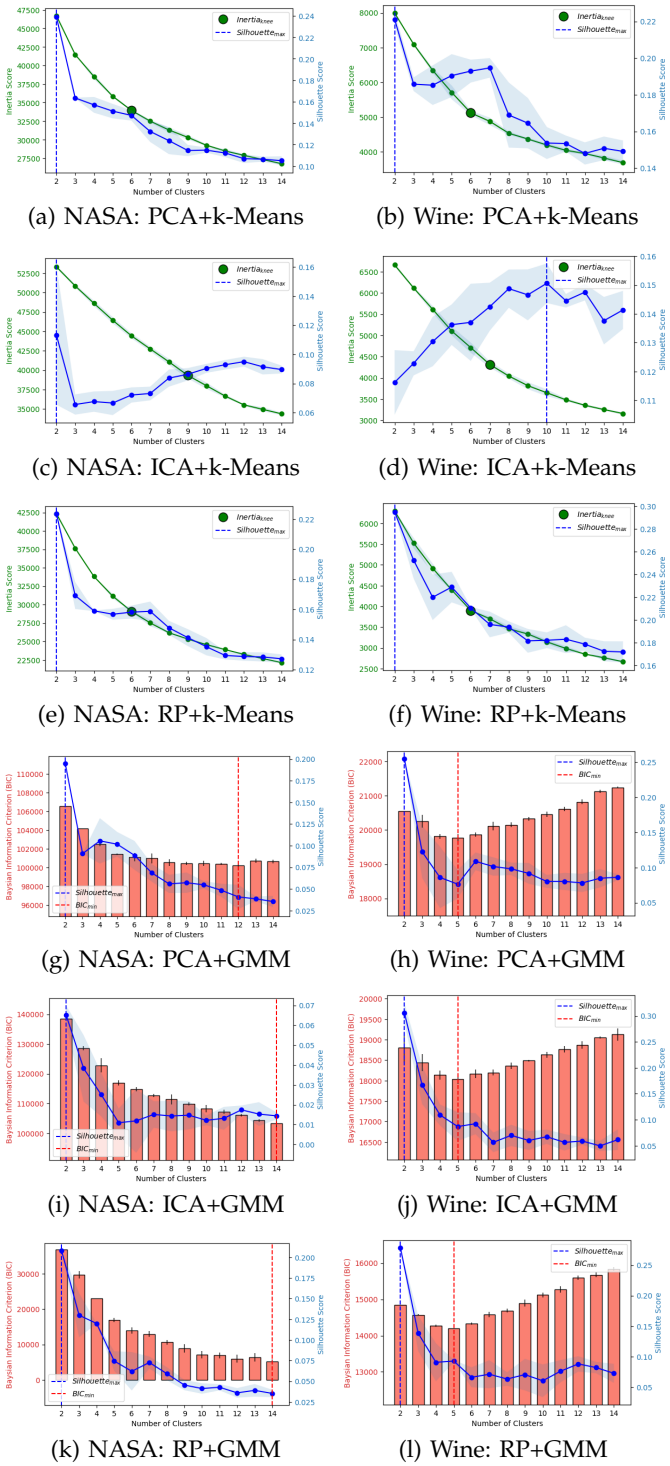
(g) NASA: RP (ρ : 15) (h) Wine: RP (ρ : 9)

Fig. 4: Dataset reduction correlation analysis

explored. The summarized results can be seen visualized in 5.

A. k-Means Combination Analysis

For the NASA dataset, PCA combined with k-means clustering shows optimal clusters around 2, supported by silhouette scores, though the inertia does not indicate a clear knee point. ICA combined with k-means similarly suggests 2 clusters with some variability. RP with k-means also indicates a stable clustering trend around 2 clusters despite some variability introduced by randomness. In the Wine dataset, PCA with k-means indicates primary clustering around 2 clusters with a secondary peak at 6-7, aligning well with the dataset's true structure. ICA with k-means suggests a higher range of 8-10 clusters, reflecting the capture of more independent components. RP with k-means also suggests 2 clusters, but the randomness affects the stability of these



(a) NASA: PCA+k-Means (b) Wine: PCA+k-Means

(c) NASA: ICA+k-Means (d) Wine: ICA+k-Means

(e) NASA: RP+k-Means (f) Wine: RP+k-Means

(g) NASA: PCA+GMM (h) Wine: PCA+GMM

(i) NASA: ICA+GMM (j) Wine: ICA+GMM

(k) NASA: RP+GMM (l) Wine: RP+GMM

Fig. 5: Combined dimensionality reduction and clustering analysis for optimal k selection using (Blue) Silhouette score, (Green) Interia, and (Red) BIC.

results. Overall, depending on the feature combinations, an impact on the number of clusters is observed. Thus, the structure of the data is susceptible to the approach.

B. GMM Combination Analysis

For the NASA dataset, PCA combined with GMM shows an optimal cluster range of either 2 or 12, depending on the selected metric. ICA with GMM shows

the kurtosis metric suggesting 2 clusters, but the BIC indicates overfitting with 13-14 clusters. RP with GMM follows a similar trend to ICA, with some noise from randomness affecting the results. In the Wine dataset, PCA with GMM shows a stable peak around 5 clusters, with the BIC metric indicating good performance across combinations. ICA with GMM suggests 2 clusters according to the kurtosis metric, which does not align well with the true structure, indicating less effective performance. RP with GMM also incorrectly suggests 2 clusters using kurtosis, with the randomness affecting the BIC metric, leading to less accurate cluster identification. Regardless of method combinations, the overall clusters do not show large changes. As such, the inherent data structure is the dominant aspect. This could be due to an excess of data noise.

In general, the following observations of the findings can be summarized for the explored methods:

- *PCA:* Consistently performs well with both k-means and GMM across both datasets. It effectively reduces dimensionality while preserving the variance, leading to stable and accurate cluster identification.
- *ICA:* Shows variability, especially with GMM on the Wine dataset. While it captures independent components, it may not always align with the optimal clustering structure.
- *RP:* Introduces randomness, leading to variability in results. It performs reasonably well with k-means on the NASA dataset but shows less stability with GMM, especially on the Wine dataset. While effective for extremely large feature sets, smaller sets may not exhibit clear benefits.

Specific properties of the data influence the outputs of various algorithms. High-dimensional data can lead to the curse of dimensionality, affecting algorithms differently. PCA performs well in reducing dimensions, while RP offers faster but potentially less precise reductions. Sparse data benefits from PCA and ICA, which capture variance and independence, respectively, whereas RP might introduce artifacts if not carefully tuned. Non-Gaussian distributions favour ICA, which maximizes non-Gaussianity, while PCA, assuming linearity, may struggle with non-linear structures without kernel transformations. High collinearity impacts PCA as redundant features contribute to explained variance; ICA may better separate independent components, whereas RP's performance varies with randomness and projection directions.

VI. Neural Network Analysis

This study employs a methodology inspired by [1], [2], implementing a hybrid training strategy that combines data standardization, 5-fold cross-validation, and 80/20 hold-out test evaluations to ensure a reliable assessment of each neural network model's performance. Given the moderate size of the datasets, a stratified sampling approach is employed throughout the training and cross-validation procedures to preserve class distribution within the testing sets. The optimal model configurations are determined via the grid search approach defined in Table I. The optimal hyper-parameter results for each evaluated model and corresponding datasets are summarized in Table II.

TABLE I: Summary of grid search hyper-parameters

Hyperparameter	Value
MLP	
Layer Size	{1, 2, 3}
Number Nodes	{5, 10, 15, 20, 25}
Epochs	{50, 100}
Activation Functions	{Tanh, Relu, Sigmoid}
Optimizer	{ADAM, SGD}

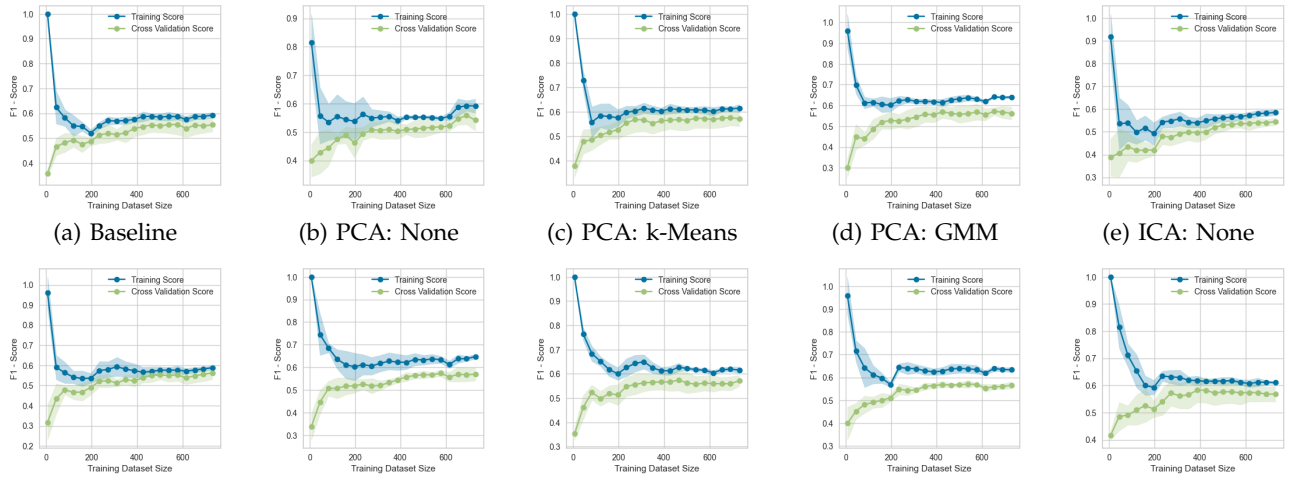
The optimal neural network model previously determined using backpropagation from study [1] is used as a benchmark to ensure a fair comparison. The datasets reduced using PCA, ICA, and RP are assessed separately. Additionally, datasets augmented with clusters derived from k-means and GMM are also evaluated. The clustering labels appended to the original and reduced datasets serve as additional features, potentially enhancing the neural network's ability to discern patterns in the data.

TABLE II: Grid search hyper-parameter optimal results using 5-Fold cross-validation

MLP Baseline	
MLP	{#F: 11, Node (Layer): [5, 20, 20] (3), Acti: Tanh, Epoch:100, Opti:ADAM}
No Clustering	
PCA	{#F: 9, Node (Layer): [25, 5, 20] (3), Acti: Relu, Epoch:100, Opti:ADAM}
ICA	{#F: 8, Node (Layer): [15, 20, 25] (3), Acti: Relu, Epoch:50, Opti:ADAM}
RP	{#F: 9, Node (Layer): [20, 25, 15] (3), Acti: Tanh, Epoch:100, Opti:ADAM}
k-Means Clustering	
PCA	{#F: 10, Node (Layer): [25, 10, 10] (3), Acti: Tanh, Epoch:100, Opti:ADAM}
ICA	{#F: 9, Node (Layer): [5, 15, 15] (3), Acti: Tanh, Epoch:100, Opti:ADAM}
RP	{#F: 10, Node (Layer): [20, 25, 15] (3), Acti: Relu, Epoch:100, Opti:ADAM}
GMM Clustering	
PCA	{#F: 10, Node (Layer): [15, 15, 25] (3), Acti: Relu, Epoch:100, Opti:ADAM}
ICA	{#F: 9, Node (Layer): [15, 25, 20] (3), Acti: Relu, Epoch:100, Opti:ADAM}
RP	{#F: 10, Node (Layer): [25, 25, 15] (3), Acti: Tanh, Epoch:100, Opti:ADAM}

TABLE III: Final test set evaluated results

No Clustering				
Metrics	Baseline	PCA	ICA	RP
Accuracy	0.65	0.63	0.61	0.65
Precision	0.61	0.58	0.60	0.62
Recall	0.65	0.63	0.61	0.65
F1-score	0.63	0.59	0.58	0.63
k-Means Clustering				
Metrics	Baseline	PCA	ICA	RP
Accuracy	0.65	0.65	0.59	0.62
Precision	0.62	0.61	0.55	0.59
Recall	0.65	0.65	0.58	0.62
F1-score	0.63	0.63	0.57	0.60
GMM Clustering				
Metrics	Baseline	PCA	ICA	RP
Accuracy	0.65	0.65	0.63	0.65
Precision	0.62	0.61	0.60	0.62
Recall	0.65	0.65	0.63	0.65
F1-score	0.63	0.63	0.61	0.63



(a) Baseline (b) PCA: None (c) PCA: k-Means (d) PCA: GMM (e) ICA: None

(f) ICA: k-Means (g) ICA: GMM (h) RP: None (i) RP: k-Means (j) RP: GMM

Fig. 6: Neural network learning curves for varying clustering and dimensionality reduction methods

A. Neural Network Analysis and Comparison

The final evaluation of the models on the hold-out test dataset provides insights into their performance on the Wine dataset. The key metrics considered include accuracy, precision, recall, and F1-score. Table III summarizes the corresponding prediction results. The learning curves for the various algorithms are analyzed in Figure 6. Each curve shows the F1-score performance metric for training and validation datasets as a function of training dataset size. The multi-class predictions consider a weighted average, which is an appropriate metric when the dataset is imbalanced [9].

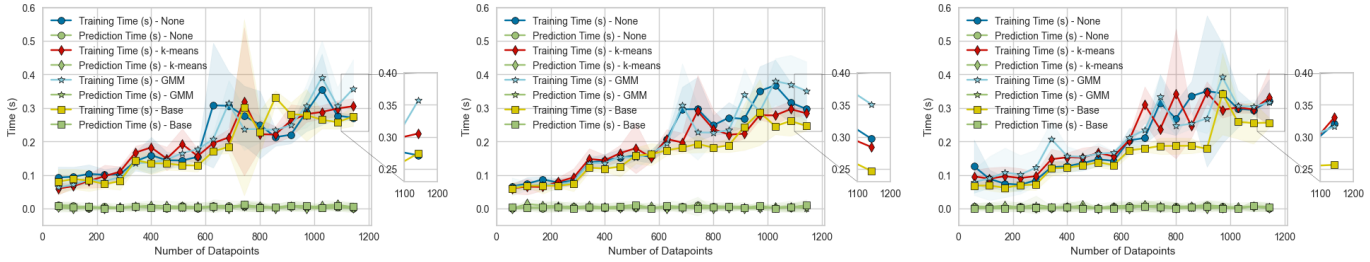
- *PCA:* Without clustering, shows a slight decrease in accuracy and F1-score compared to the baseline, with an improvement in recall but a drop in precision. This indicates that while PCA effectively captures variance, it might lead to overfitting. However, when combined with k-means clustering, PCA maintains the same accuracy and F1-score as the baseline, with a slight improvement in precision. This combination suggests a balanced bias-variance trade-off, enhancing generalization without performance loss. PCA with GMM clustering achieves the same metrics as the baseline, indicating that GMM clustering helps stabilize performance.
- *ICA:* Without clustering shows a decrease in all metrics compared to the baseline, highlighting its higher variance and potential misalignment with the true data structure. Combining ICA with k-means clustering further deteriorates performance, suggesting that k-means does not effectively complement ICA. When combined with GMM clustering, ICA shows slight improvements in precision and recall compared to no clustering, though it still underperforms relative to PCA and the baseline. This indicates some benefit from GMM clustering, but the improvement is minimal.
- *RP:* Without clustering, it maintains similar metrics to the baseline, indicating its effectiveness despite its inherent randomness. However, RP combined with k-means clustering shows a slight decrease in accuracy and F1-score, with minimal changes in precision and recall. This suggests that the randomness of RP, combined with k-means clustering, might introduce instability. Conversely, RP with GMM clustering achieves a performance similar to the baseline, indicating that GMM clustering helps stabilize the variability introduced by RP.

Overall, it is observed that the learning curves of each method generally follow a similar trend. However, upon further investigation, adding k-means and GMM clustering labels generally enhances model performance across all dimensionality reduction methods by providing additional structural information that aids in better generalization. PCA, especially with clustering methods, provides the most effective dimensionality reduction, leading to consistent neural network performance on the reduced datasets. ICA and RP, while useful, require additional tuning and might not capture the optimal data structure as effectively as PCA. This analysis underscores the importance of selecting appropriate dimensionality reduction and clustering techniques to improve model generalization and performance, with PCA and clustering, particularly k-means, proving to be the most reliable method for achieving a balanced bias-variance trade-off.

B. Model Time Complexity

In addition to prediction performance, time performance is a crucial metric when comparing the algorithms. The training and prediction times, summarized for both datasets in Figure 7, provide valuable insights into the efficiency of the models.

The analysis of training times reveals that PCA and ICA experience slightly higher training times than the baseline as data points increase, with additional clustering methods (k-means and GMM) further elevating these times due to their computational overhead. Nevertheless, PCA with clustering maintains reasonable training efficiency. ICA shows a more noticeable increase in



(a) PCA (b) ICA (c) RP

Fig. 7: Neural network time comparison for various dimensionality reduction and clustering methods

training times than PCA, particularly when clustering is added, reflecting ICA's complexity in extracting independent components. RP exhibits higher variability in training times due to its randomness, with clustering methods, especially GMM, increasing training times further. Despite this, RP's training times without clustering remain comparable to those of PCA and ICA. Prediction times across all methods are relatively stable and low, indicating that the impact of dimensionality reduction and clustering is more significant during training rather than prediction. Unexpectedly, reducing the number of features does not necessarily result in faster training times. Time savings are only significant when a substantial number of features are reduced. The baseline model benefits from having the fewest nodes, and while adding a feature does increase training time slightly, the overall impact is minimal. This underscores the importance of balancing feature reduction with model complexity to achieve optimal training efficiency.

VII. Conclusion

This study demonstrates the effectiveness of various clustering and feature reduction techniques on datasets with a moderate number of features (10 & 17) and classes (2 & 6). Our findings indicate that the specific dataset's characteristics significantly influence these methods' success. This observation aligns with the no-free-lunch theorem [8], which suggests that no single method is best for all problems. A general summary and hypothesis confirmation (✓) or rejection (×) is indicated in Table IV.

Overall, the limitations suggest that the current feature set might be too small to leverage significant computational improvements through feature reduction techniques alone. Additionally, the clustering results were impacted by the datasets' noise and the metrics' suitability. This highlights the need for more rigorous data pre-processing, including outlier detection and noise reduction, to enhance clustering performance. Therefore, implementing more advanced data-cleaning techniques could improve the quality of the clustering outcomes in future studies. Additionally, exploring alternative metrics and more robust clustering algorithms might provide better insights, particularly for datasets with higher noise levels. Nevertheless, these methods have demonstrated that data can be successfully clustered and transformed into lower-dimensional feature sets with little to no loss of information.

TABLE IV: Model result and hypothesis summary

Models	Pros	Cons
kMeans	✓ Demonstrates relatively fast convergence when compared to GMM • Maintains performance metrics and improves precision when combined with PCA.	✓ Sensitive to point initialization. • Does not complement ICA effectively, leading to performance deterioration. • Increases training times when combined with dimensionality reduction methods.
GMM	✓ Provides additional structural information that helps better performance and generalize the data.	✓ Does not significantly alter performance metrics but increases training times.
PCA	✓ Maintains reasonable training times and performance metrics when combined with clustering methods. • Captures underlying data structure effectively, reducing overfitting. • Consistently reduces the gap between training and cross-validation scores.	× Slight decrease in performance metrics without clustering, indicating possible overfitting on training data. (И: Expected to be best performing in NN) • Introduction of clustering methods increases training times.
ICA	• Suitable for datasets where components are statistically independent. • Shows slight enhancement in precision and recall when combined with GMM.	✓ Exhibits higher variance and misalignment with the true data structure. ✓ Underperforms compared to PCA and the baseline, particularly with k-means clustering.
RP	✓ Maintains low and stable prediction times across different dataset sizes. × Comparable training times to PCA and ICA without clustering, effective in high-dimensional data. (И: Expected to be fastest)	✓ Slight decrease in performance metrics with clustering, indicating potential instability. ✓ Clustering methods, particularly GMM, lead to higher training times. • Introduces variability due to randomness, leading to higher variance in training times.

VIII. Resources

[1] *Supervised Learning.* Odendaal, K. (2024).
[2] *Random Optimization.* Odendaal, K. (2024).
[3] *API Reference.* Scikit-learn. https://scikit-learn.org.
[4] *API Reference.* Scipy. https://scipy.org.
[5] *Machine Learning LaTeX Template.* Nakamura, K. (2023).
[6] *Nasa Asteroid Dataset.* Kaggle. https://www.kaggle.com/datasets/shrutimehta/nasa-asteroids-classification.
[7] *Wine Quality Dataset.* Kaggle. https://www.kaggle.com/datasets/yasserh/wine-quality-dataset.
[8] *Machine Learning.* Mitchell, T. M. vol. 1, (1997).
[9] *Hands-on machine learning with Scikit-Learn, Keras and TensorFlow: concepts, tools, and techniques to build intelligent systems.* Geron, A. 2nd ed, (2019).