Erasmus University Rotterdam
Erasmus School of Economics
Bachelor Thesis Econometrie en Operationele Research

A Comparative Analysis of Clustering Quality Based on Internal Validation Indices Using Datasets with Different Characteristics

Gwen de Haas (544360)

Supervisor:	dr. Marina Khismatullina
Second assessor:	drs. Jeffrey Durieux
Date:	1st July 2023

Abstract

In this paper, we investigate the impact of four distinct dimensionality reduction methods on the cluster quality for two clustering algorithms. To accomplish this, we employ internal clustering validation indices as a means of evaluation. The study explores both linear techniques, such as Principal Component Analysis (PCA) and Independent Component Analysis (ICA), and non-linear methods, including t-Distributed Stochastic Neighbor Embedding (t-SNE) and Locally Linear Embedding (LLE). Evaluation of partitioning and hierarchical clustering is conducted using k-means and agglomerative hierarchical clustering (AGNES) algorithms. Previous research by Renjith, Sreekumar and Jathavedan (2021) investigated this topic using the Jester 1 dataset, which consisted of 100 joke ratings. Their findings suggested that t-SNE outperformed other methods when combined with k-means and AGNES. However, Renjith et al. (2021) emphasized the significant impact of the nature of the dataset on method performance. In this paper, we extend the investigation of Renjith et al. (2021) by using two additional datasets: a financial dataset comprising financial firm ratios and a biology dataset containing codon frequencies. These datasets differ notably in nature, particularly regarding variable dependencies. Our findings reveal that the results reported by Renjith et al. (2021) are altered when using these alternative datasets, with t-SNE no longer demonstrating superior performance. Significantly, we observe that when there are more variables with stronger dependencies, the preservation or appropriate handling of these relationships becomes increasingly critical for dimensionality reduction techniques.

1 Introduction

In this paper, we investigate the impact of four distinct dimensionality reduction methods on the cluster quality for two clustering algorithms. To accomplish this, we employ internal clustering validation indices as a means of evaluation.

Dimensionality reduction is a technique that reduces the number of features in a dataset while retaining important information (Assent, 2012). It simplifies high-dimensional data by mapping it to a lower-dimensional space (Van Der Maaten, Postma, Van den Herik et al., 2009), resulting in noise reduction, identification of relevant features, and computational efficiency improvement (Ding, He, Zha & Simon, 2002).

Clustering, on the other hand, groups similar data points together without prior knowledge of their labels or target variable, aiming to discover patterns and natural groupings within the dataset (Madhulatha, 2012).

Performing dimensionality reduction before clustering is advantageous as it leads to improved cluster quality (Huang, Wu & Ye, 2019) and enhances data interpretability and visualization (J. Tang, Liu, Zhang and Mei (2016)).

The study conducted by Renjith et al. (2021) examine the impact of four distinct dimensionality reduction methods on cluster quality, utilizing internal clustering validation indices. Their selection of dimensionality reduction techniques include both linear methods such as Principal Component Analysis (PCA) and Independent Component Analysis (ICA), as well as non-linear methods like t-Distributed Stochastic Neighbor Embedding (t-SNE) and Locally Linear Embedding (LLE). Linear dimensionality reduction techniques preserve linear relationships between the variables and employ linear transformations to reduce the dimensionality of the data, while non-linear dimensionality reduction techniques employ non-linear transformations to map the data from a high-dimensional space to a lower-dimensional space and capture non-linear relationships and complex patterns, allowing for a more flexible representation of the data.

To evaluate the performance of these dimensionality reduction techniques, Renjith et al. (2021) employ k-means clustering and agglomerative hierarchical clustering (AGNES). Their findings demonstrate that t-SNE performs best when combined with k-means and AGNES. However, the researchers emphasize that the nature of the dataset strongly influenced the performance of the methods.

The Jester dataset analyzed in the study of Renjith et al. (2021) comprised subjective ratings for 100 jokes. This dataset falls under the category of social datasets and is characterized by self-reported data, a specific number of observations, and dimensionality. Notably, the variables in this dataset do not exhibit significant dependencies. One possible explanation for this phenomenon is the subjective nature of humor. What may be amusing to one person may not elicit the same response from another, leading to an absence of consistent patterns or dependencies in how jokes are evaluated by different individuals. Consequently, it becomes crucial to investigate the impact of utilizing datasets with diverse characteristics on the obtained results. As highlighted by Kwon and Sim (2013), dataset properties such as domain area, dimensionality, ratio of missing values, class dimensionality, and functional dependency of features influence the performance of clustering methods. This leads us to the research question: *How do the results of cluster quality vary across the four distinct dimensionality reduction techniques when datasets*

with different characteristics are employed?

To address this research question, we extend the investigation to two additional datasets: a financial dataset comprising financial ratios of firms and a biology dataset containing codon frequencies. These datasets differ significantly in nature, particularly in terms of variable dependencies. Unlike the Jester dataset, the financial dataset reveals specific variables with notable positive or negative dependencies on other variables. Similarly, the biology dataset exhibits a high level of dependency among nearly all variables.

Our analysis reveals that t-SNE no longer demonstrates superior performance for the financial and biology datasets. Instead, LLE emerges as a strong contender for the financial dataset, while ICA performs well for the biology dataset. The different result between these two datasets could be due to the fact that the relationship between the variables in the financial dataset are more non-linear and that of the biology dataset is linear, resulting in a non-linear and linear dimensionality reduction technique respectively. Furthermore, as the biology dataset contains more highly dependent variables, this might make it more important that the resulting dataset is independent, whereas for the financial dataset it is enough that the dependencies are retained. Nonetheless, these results emphasize the importance of preserving and appropriately handling dependencies when variables exhibit substantial relationships.

To the best of our knowledge, the research conducted by Renjith et al. (2021) has not been applied to real datasets with different characteristics. There has however been previous research on the combined effect of dimensionality reduction techniques and clustering accuracy on a single real dataset or datasets with the same characteristics. Song, Yang, Siadat and Pechenizkiy (2013) performed a comparative study of dimensionality reduction techniques, including PCA, on the performance of clustering on a dataset with real event log recorded from patient treatment that indicated that PCA with k-means was most suitable for the complex dataset. Additionally, B. Tang, Shepherd, Milios and Heywood (2005) conducted a comparison of six dimensionality reduction methods, including PCA and ICA among others, in the context of text clustering. In this study, it became clear that ICA ranked the highest for classification accuracy and stability. According to these studies, ICA and PCA seem to perform very well for clustering. However, the studies conducted by B. Tang et al. (2005) and Song et al. (2013) did not incorporate the dimensionality reduction methods, LLE and t-SNE, that we specifically utilize in this paper.

This paper could have a significant impact as it highlights which methods work better with specific types of data. This information can be valuable for researchers and companies who can then choose the appropriate methods to interpret their data accurately. Ultimately, this can lead to better insights and decision-making based on the data analysis.

The paper is organized as follows: Section 2 provides an overview of relevant literature related to our study. In Section 3, we describe the data used and outline the data cleaning process. Section 4 discusses the methods employed in our analysis. The results of our study are presented in Section 5. Finally, in Section 6, we summarize and discuss our main findings.

2 Theoretical Framework

Renjith et al. (2021) concluded that the performance of the dimensionality reduction technique and the clustering algorithms are significantly influenced by the nature or characteristics of the

dataset. The paper by Kwon and Sim (2013) identified these influential characteristics of the dataset which are presented in Table 1.

Table 1: Data characteristics

Characteristics elements	Description
Sample size	Sample size for learning and testing number of instances (Smaller or larger)
Class type	Binary or multiple classes
Missing values (sparse data)	Ratio of instance which has missing values
Functional dependency of features	Total degree of functional dependency between features whether there is degree of dependency or not
Dimensionality (number of features)	The number of features low dimensional or high dimensional
Domain area	Social (noisy), natural science (straightforward)
Continuous feature	Ratio of continuous features per nominal feature
Class dimensionality	The number of features consisting class single class or multi class

Notes: Reprinted from Kwon and Sim (2013)

Previous research has investigated the impact of certain characteristics of datasets on the performance of clustering algorithms and dimensionality reduction techniques. For instance, Renjith, Sreekumar and Jathavedan (2020) conducted a study where they varied the cardinality and dimensionality parameters of the dataset to assess their influence on the performance of various clustering algorithms. Their findings indicated that k-means produced the best results when the cardinality was varied, while changes in dimensionality did not significantly affect hierarchical clustering. Additionally, Mohamad and Usman (2013) highlighted that the presence of features with substantial size or variability in a dataset can significantly impact clustering outcomes. These studies are particularly relevant to our research as we will be working with different datasets that vary in dimensionality and may contain features with significant size or variability. Therefore, it is plausible that these differences in characteristics may influence the overall clustering results obtained in the study conducted by Renjith et al. (2021).

The evaluation conducted by Fernández, Javier, Verleysen, Lee and Ignacio (2013) focused on assessing the stability, robustness, and performance of different dimensionality reduction techniques. The comparison included methods such as LLE, t-SNE, and PCA. The study revealed that LLE is more sensitive to small changes in data and parameter variations. However, as the dataset size increases, the influence of these parameters diminishes. Additionally, LLE tended to produce non-fully connected graphs, leading to improper embedding of some data points, whereas PCA and t-SNE were preferred for data visualization. Based on these findings, it is plausible to anticipate that LLE may yield less satisfactory cluster representations, especially when applied to datasets with a lower number of variables. This expectation is particularly relevant to our two additional datasets, as they possess a smaller number of variables compared to the original Jester dataset.

In a related study, Zubova, Kurasova and Liutvinavičius (2018) investigated the accuracy of

dimensionality reduction techniques such as PCA, ICA, and LLE using non-clustered randomly generated data, clustered randomly generated data, and real data. The study revealed that the accuracy of these techniques remained unchanged with an increase in the number of observations. However, it was observed that higher dimensionality resulted in lower accuracy. Considering that our additional datasets will have lower dimensionality compared to the original Jester dataset, we can expect, based on these findings, that the overall accuracy for the additional datasets will be higher.

These variations in accuracy observed across different dimensionality reduction techniques have significant implications for clustering accuracy. Dimensionality reduction techniques play a vital role in reducing noise, eliminating irrelevant or redundant features, and uncovering underlying patterns within the data (Huang et al., 2019). As a result, the choice of dimensionality reduction method can greatly influence the overall quality and reliability of clustering results.

Moreover, datasets from different domains can exhibit varying degrees of variable dependency and patterns. The dimensionality reduction techniques discussed in the study conducted by Renjith et al. (2021) address dataset dependency in distinct ways. PCA produces uncorrelated components, while ICA ensures independent components as output. LLE and t-SNE, although not explicitly addressing variable dependency, preserve the global structure of the data, which can capture dependencies among variables. In the original Jester dataset used by Renjith et al. (2021), the dependency among variables was not particularly strong, so the importance of addressing this dependency may have been less significant, and the preservation of the global structure achieved by t-SNE was deemed sufficient. However, in our additional datasets, where stronger dependencies among variables are evident, it becomes more crucial to address these dependencies appropriately. Anowar, Sadaoui and Selim (2021) compared dimensionality reduction techniques using various measures, including correlation, and found that non-linear techniques outperformed linear techniques in capturing correlation and other factors. The presence of variable dependency within the resulting dataset can subsequently influence the outcomes of clustering (Sambandam, 2003). In cases where dependent variables are used, certain variables may carry more weight than others, potentially leading to biased interpretations (Sambandam, 2003). Therefore, for our additional datasets exhibiting stronger dependencies between variables, LLE and t-SNE may yield better cluster quality results based on these studies.

Considering these studies collectively, it becomes apparent that modifying multiple characteristics of the data, including dimensionality and variable dependencies, will significantly impact the overall outcomes of the study conducted by Renjith et al. (2021).

3 Data

3.1 Data exploration

In this study we continue to employ the Jester 1 dataset (Goldberg, Roeder, Gupta & Perkins, 2001), which comprises ratings ranging from -10 to +10 associated with 100 jokes. This dataset, sourced from social media, includes a total of 73,421 observations and contains a substantial amount of missing data. As mentioned earlier, it was originally used in the comparative study conducted by Renjith et al. (2021).

In order to evaluate the impact of different dataset characteristics on the findings of Renjith et al. (2021), we will investigate two additional datasets. The first dataset used in our analysis is the Financial Ratios Firm Level dataset from Wharton Research Data Services (2023). This dataset consists of 572,941 U.S. companies, including instances with missing values, and encompasses 63 variables related to various financial aspects such as financial soundness, liquidity ratios, valuation ratios, profitability ratios, capitalization ratios, solvency ratios, and efficiency ratios.

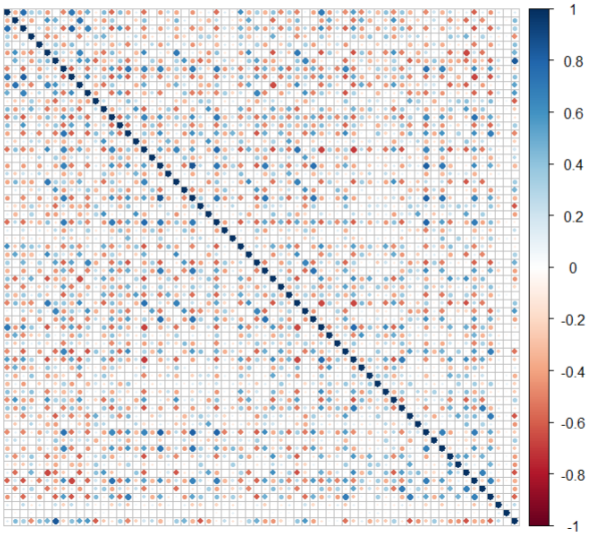
It is worth noting that these ratios can vary depending on the size of the firms. Larger firms may exhibit different ratios compared to smaller ones due to factors such as economies of scale, resource access, and market power. Furthermore, the financial ratios can also differ based on the industry characteristics of the firms. For instance, capital-intensive industries like manufacturing or utilities may display different leverage ratios compared to service-based industries. Similarly, sectors with high research and development expenditures, such as technology or pharmaceuticals, may exhibit distinct profitability ratios. Moreover, competitive dynamics and market conditions play a role in shaping financial ratios across sectors. Some sectors may experience intense competition and lower profit margins, while others may have concentrated markets and greater pricing power. These factors impact financial ratios like gross margin and market valuation ratios. Considering these various factors, it is expected that the dataset will reveal clusters of firms with similar ratios based on their specific characteristics as outlined above.

Although this particular dataset has not been previously employed in research, the study conducted by Zubova et al. (2018) offers insights into the dimensionality reduction methods applied to financial ratios. Their research compared PCA, ICA, and LLE, indicating that LLE yielded the least favorable outcomes while PCA and ICA performed better. However, the rationale behind these results was not explicitly provided.

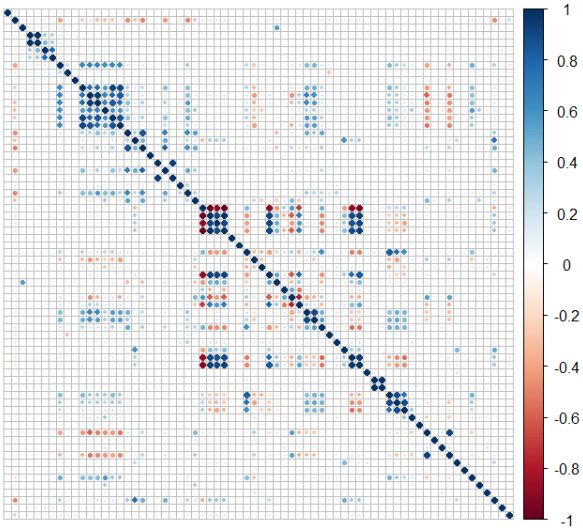
The second dataset analyzed in this study originates from the publicly available Codon Usage Tabulated from Genbank (CUTG), accessible through the UCI Machine Learning Repository (Dua & Graff, 2017). This dataset focuses on the coding DNA and encompasses a diverse array of organisms from various categories. In the realm of genetics, a codon represents a sequence of three nucleotides that carries genetic information, encoding amino acids or signaling the termination of protein synthesis. Comprising 13,028 observations, with some instances containing missing values, the dataset comprises 64 variables (National Human Genome Research Institute, 2023). Due to the distinct codon frequencies found in different organisms (Athey et al., 2017), it is anticipated that this dataset will exhibit clusters, wherein organisms with similar codon frequencies are grouped together. Given the potential variability in the importance of different codons, performing dimensionality reduction on this dataset proves beneficial. For instance, certain codons may be rare and occur exclusively in specific organisms, while others occur frequently and are present in all organisms, such as stop and start codons that indicate protein synthesis termination or initiation. Previous studies, such as that conducted by Khomtchouk (2020), have utilized this dataset to investigate the grouping of specific combinations of codons, known as genetic code units. Their research demonstrated that analyzing the frequencies of codon usage offers a valuable approach for classifying DNA and predicting the taxonomic identity of organisms.

In addition to variations in sample size, dimensionality, and domain area among the three

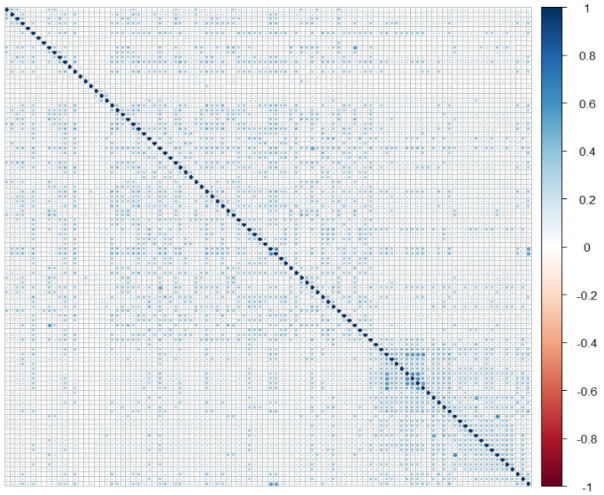
analyzed datasets, there are also differences in the dependency among their variables. The correlation plots can be found in Figure 1.



(a) Biology correlation plot



(b) Financial correlation plot



(c) Jester correlation plot

Figure 1: Correlation plots (**a**) biology dataset (**b**) financial dataset (**c**) Jester 1 dataset.

In relation to the Jester dataset, it is worth noting that most variables exhibit a certain level of dependence on other variables, although this dependency is not particularly strong. In contrast, the financial dataset unveils specific variables that showcase a notable positive or negative dependency on other variables. The degree of dependency tends to be more evident among variables that are closely located within the columns of the dataset, which could be attributed to these variables belonging to the same category or exhibiting similar characteristics. For instance, we observe a high positive correlation between the variables "cash_ratio" and "quick _ratio" both of which fall under the category of liquidity ratios. Similarly, the variables "equity_invcap," "debt_invcap," and "totdebt" exhibit a strong negative correlation, all belonging to the category of capitalization ratios. On the other hand, the biology dataset demonstrates a substantial level

of dependency among nearly all variables, regardless of whether the dependency is positive or negative. This significant dependency can be attributed to codon usage bias, which involves the preferential or non-random utilization of synonymous codons (Parvathy, Udayasuriyan & Bhadana, 2022). As discussed in the Section 2, the dimensionality reduction techniques employed in this study address dependencies in distinct ways. Therefore, the divergent patterns of dependency observed in the datasets have the potential to impact the findings of Renjith et al. (2021).

3.2 Data cleaning

In our study, each of the three datasets analyzed contains missing values, although the extent of missingness varies among them. To handle this issue, we have decided to remove all observations with missing values from the datasets. Specifically, the financial dataset initially consists of 572,941 observations, out of which 525,017 observations contain missing values. After removing the missing values, the dataset is reduced to 47,924 observations. Similarly, the biology dataset has a total of 13,028 observations, with only 2 observations containing missing values. Therefore, after removing the missing values, the dataset is left with 13,026 observations. Lastly, the Jester dataset contains 73,421 observations, with 59,305 observations containing missing values. After removing the missing values, the dataset is reduced to its remaining 14,116 observations.

Furthermore, in addition to addressing missing values, we also eliminate duplicate observations from the datasets to enhance data consistency. Initially, the financial dataset contained 41 duplicate entries, resulting in 47,924 unique observations. Similarly, the biology dataset had 42 duplicates, leaving 13,026 unique observations. On the other hand, the Jester dataset did not contain any duplicate entries, resulting in 14,116 unique observations. By removing both missing values and duplicates from the datasets, we guarantee that the data used in our study is clean and devoid of such discrepancies. This process ensures the reliability and accuracy of our analyses.

To align with the methodology employed by Renjith et al. (2021) and minimize potential influences stemming from dataset size variations, we opt to randomly sample 5,000 observations from each dataset for further analysis. This sample size choice mirrors the approach taken by Renjith et al. (2021) in their visualization and likely in their analysis. Although Renjith et al. (2021) did not explicitly mention setting a seed value, we take the precautionary measure of establishing a seed value to ensure the reproducibility of the random sampling process throughout our analysis.

4 Methodology

This paper will follow the methodology presented by Renjith et al. (2021), which involves using either a linear or non-linear dimensionality reduction technique on the data (Section 4.1), followed by two clustering algorithms (Section 4.2). The quality of the clusters will then be evaluated using four different internal validation indices (Section 4.3). To carry out these methods, multiple R packages are utilized which are listed in Appendix C.

4.1 Dimensionality reduction techniques

We will use four techniques for reducing the dimensionality of the datasets, including two linear dimensionality reduction methods: principle component analysis (Section 4.1.1) and independent component analysis (Section 4.1.2) and two non-linear dimensionality reduction methods: t-Distributed Stochastic Neighbor Embedding (Section 4.1.3) and Locally Linear Embedding (Section 4.1.4).

4.1.1 Principal component analysis (PCA)

Principal component analysis (PCA) is a linear technique that reduces the dimensionality of a dataset, capturing as much variation as possible (Groth, Hartmann, Klie & Selbig, 2013). By retaining the essential information of the data while reducing its size, the dataset is simplified making it easier to interpret and analyze (Abdi & Williams, 2010).

The first step in computing the principal components (PC's) involves calculating the symmetric covariance matrix, denoted as S, for the data matrix X (refer to Equation 1 as presented by Johnson and Wichern (2014)).

$$S = \frac{1}{p} \times (X - \mu)(X - \mu)^T \tag{1}$$

Here, p is the amount of variables, X represents the matrix containing the data, μ denotes the mean vector of the variables, and the superscript T signifies the transpose operation.

Using S, we can proceed to compute the eigenvalues and eigenvectors utilizing the approach outlined by Poole (2015). The process begins by applying Equation 2a, which can be alternatively expressed as Equation 2b. In this equation, I represents the identity matrix. To obtain a non-trivial solution $v \neq 0$ and satisfy the condition that the determinant of the matrix (A - $\lambda \times$ I) equals zero, we formulate Equation 2c. By solving Equation 2c, we can determine the eigenvalues, λ, and subsequently obtain the eigenvectors, v.

$$\begin{aligned} a: &\quad S \times v = \lambda \times v \\ b: &\quad (S - \lambda \times I) \times v = 0 \\ c: &\quad \det(S - \lambda \times I) = 0 \end{aligned} \tag{2}$$

Based on the eigenvectors, we can calculate the principal components, denoted as Z_m. Each principal component is a linear combination of the original variables x_i, where the weights are given by the corresponding entries in the eigenvectors (see Equation 3 by James, Witten, Hastie and Tibshirani (2013)).

$$Z_m = \sum_{i=1}^{p} v_{im} x_i \tag{3}$$

In Equation 3, v_{im} represents the ith element of the eigenvector m. The number of principal components, m, is less than the number of variables, p.

Important characteristic of the PC's is that they are orthogonal to each other (uncorrelated) and that they maximize the variability of the linear combinations. However, the appropriate number of principal components, m, to retain is often a non-trivial matter that requires the use

of methods such as cross-validation or scree plots (Johnson & Wichern, 2014). In this paper we use the scree plot and elbow method to determine the optimal number of principal components to retain. The scree plot presents the eigenvalues of the principal components and helps identify the point at which the eigenvalues exhibit a significant drop (the elbow point), indicating the number of components to be retained. This plot serves as an indicator of the ideal number of components to retain by visually identifying the point where the eigenvalues decrease rapidly.

4.1.2 Independent component analysis (ICA)

Independent component analysis (ICA) is another linear dimensionality reduction method and is considered an extension of PCA. While PCA aims to find uncorrelated components, referred to as principal components, ICA seeks to identify independent and non-Gaussian components, known as independent components (Ge & Song, 2007). Unlike PCA, which requires components to be orthogonal and linear combinations of the original variables, ICA is less restrictive. Additionally, ICA does not impose any specific ordering or arrangement on the resulting components (Renjith et al., 2021).

The underlying assumption of ICA is that the observed multivariate data, denoted as X, is a mixture of unknown latent variables. This assumption can be represented by Equation 4.

$$X_{(n\times m)} = A_{(n\times n)} s_{(n\times m)} \tag{4}$$

Here, we have a matrix A with unknown coefficients, of which the inverse of this matrix holds significance in our context. Additionally, we have a vector s that consists of latent variables, known as independent components. These independent components are assumed to follow a non-Gaussian distribution and are statistically independent (Oja & Hyvarinen, 2000).

To verify the assumption of non-Gaussianity and independence, we use the Shapiro-Wilk test. The Shapiro-Wilk test examines whether the data conforms to a Gaussian distribution based on both it's skewness and kurtosis (Razali, Wah et al., 2011). The test statistic for the Shapiro-Wilk test, as presented in Equation 5 (Shapiro & Wilk, 1965), is defined as follows:

$$W = \frac{(\sum_{i=1}^{n} g_i y_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{5}$$

In this equation, y_i represents the ith order statistic, $\overline{y}$ denotes the sample mean, and g_i is computed as $\frac{h^T V^{-1}}{(h^T V^{-1} T^{-1} h)^{\frac{1}{2}}}$, with h a vector of expected values of the order statistics of independent and identically distributed random variables sampled from the standard normal distribution and V is the covariance matrix of those order statistics (Razali et al., 2011). The null hypothesis of the Shapiro-Wilk test is that the data, y_i,...,y_n is Gaussian distributed.

The correlation between variables can be evaluated using the cor() function, which quantifies the magnitude and direction of the linear association among the variables. When correlation values are low, whether positive or negative, it indicates a greater level of independence.

The initial step in ICA involves preprocessing the data matrix X. This entails centering the observations by subtracting the mean and whitening the data to ensure uncorrelated components with unit variance, as outlined by Naik and Kumar (2011).

Following the preprocessing step, the objective is to estimate the independent components

which are contained in vector, s. This is accomplished by obtaining the unmixing matrix W, which is the inverse of A in Equation 4 (refer to Equation 6).

$$s_{(n\times m)} = W_{(n\times n)} X_{(n\times m)} \tag{6}$$

However, since the matrix A is unknown, various approaches can be employed to recover W. The commonly used approach is to maximize non-Gaussianity. This involves iteratively optimizing W to maximize either the kurtosis or the negative entropy of the estimated independent components.

The kurtosis measure aims to identify the optimal W that maximizes the kurtosis of the estimated independent components, while the negative entropy measure seeks to find the W that maximizes the negative entropy, as described by Tharwat (2021). In our study, we utilize the preProcess() function, which internally employs the fastICA package that utilizes the kurtosis measure.

ICA offers advantages such as its flexibility compared to PCA and the interpretability of the independent components, which correspond to meaningful features and patterns in the data (Oja & Hyvarinen, 2000). However, a limitation of ICA is the ambiguity in the scaling and order of the independent components, as they are not uniquely defined (Naik & Kumar, 2011). Additionally, ICA assumes the statistical independence of the sources, which may not always hold in real-world scenarios, potentially resulting in inaccurate separation outcomes

Similar to PCA, the scree plot and elbow method are commonly employed to determine the number of independent components (for further details on this topic, please refer to Section 4.1.1)

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

T-Distributed Stochastic Neighbor Embedding (t-SNE) is a nonlinear technique used for dimensionality reduction. It possesses the ability to capture both the local and global structures of high-dimensional data, as highlighted by Van der Maaten and Hinton (2008).

The t-SNE algorithm begins by calculating the similarity or affinity between pairs of high-dimensional data points, denoted as x_i and x_j, as shown in Equation 7 by Van der Maaten and Hinton (2008).

$$p_{ij} = \frac{exp(-\|x_i - x_j\|^2/2\sigma^2)}{\sum_{k\neq i} exp(-\|x_i - x_k\|^2/2\sigma^2)} \tag{7}$$

In the equation, p_{ij} represents the conditional probability that x_i would select x_j as its neighbor. The term σ refers to the variance of the Gaussian in the high dimensional space, and $\|\|$ denotes the Euclidean distance.

Next, a similar process is repeated for the low-dimensional counterparts of the data points, denoted as y_i and y_j, as expressed in Equation 8 by Van der Maaten and Hinton (2008).

$$q_{ij} = \frac{exp(-\|y_i - y_j\|^2)}{\sum_{k\neq i} exp(-\|y_i - y_l\|^2)} \tag{8}$$

Lastly, the algorithm minimizes a cost function known as the Kullback-Leibler divergence, which

quantifies the discrepancy between the joint probability distribution, P, in the high-dimensional space and the joint probability distribution, Q, in the low-dimensional space, as represented by Equation 9 (Van der Maaten & Hinton, 2008).

$$C = KL(P\|Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}} \tag{9}$$

In the equation, $p_{ii} = q_{ii} = 0$, $p_{ij} = p_{ji}$, and $q_{ij} = q_{ji}$ for all values of i and j.

4.1.4 Locally Linear Embedding (LLE)

Locally Linear Embedding (LLE) is a technique for reducing the dimensionality of high-dimensional data while preserving its local structure in the embedded space. It assumes that the data exhibits its smooth and non-linear characteristics. However, verifying this assumption is difficult due to the high dimensionality of the data, making it challenging to evaluate without fitting a specific model.

The main objective of the LLE algorithm is to find a low-dimensional representation of the data, denoted as Y_i. The algorithm starts by constructing a weight matrix, R_{ij}, which represents the linear reconstruction of a data point x_i based on its neighboring data points.

To determine the optimal number of neighbors used to construct each data point x_i, we utilize the calc_k() function provided by the "lle" package in R, which implements the algorithm proposed by Kayo (2006). Appendix A provides further details about this algorithm.

To obtain the weight matrix R_{ij}, we solve a constrained least-squares problem that minimizes the reconstruction error, $\varepsilon(R)$ (see Equation 10 introduced by Roweis and Saul (2000). This error measures the discrepancy between each data point x_i and its reconstructed representation using the weight matrix.

$$\varepsilon(R) = \sum_i \|x_i - \sum_j R_{ij} x_j\|^2 \tag{10}$$

After obtaining the weight matrix R_{ij}, the low-dimensional representation Y_i is computed by minimizing the cost function $\Theta(Y)$ (see Equation 11) proposed by Roweis and Saul (2000). This optimization is performed while utilizing the fixed weight matrix R_{ij} obtained from the reconstruction error minimization step.

$$\Theta(Y) = \sum_i \|Y_i - \sum_j R_{ij} Y_j\|^2 \tag{11}$$

LLE offers the advantage of preserving the local relationships between data points (de Ridder, Kouropteva, Okun, Pietikäinen & Duin, 2003). However, it may be sensitive to noise and outliers (Renjith et al., 2021), and its performance can be heavily influenced by the parameters that need to be set, potentially impacting the obtained results (De Ridder & Duin, 2002).

4.2 Clustering

4.2.1 K-means

K-means is an algorithm used for clustering data into k clusters. Its objective is to minimize the within-cluster sum of squares (WCSS) by iteratively reallocating data points between clusters, as defined in Equation 12 (Hartigan & Wong, 1979). For a detailed explanation of the k-means clustering algorithm, refer to Appendix A.

$$WCSS = \sum_{k=1}^{K} \sum_{x_i \in C_k} \|x_i - b_{C_k}\|^2 \tag{12}$$

In the given equation, each data point is represented as x_i, cluster k is denoted as C_k, while b_{C_k} represents the centroid of cluster C_k, which acts as a representative point that effectively summarizes the characteristics of the data points within that particular cluster. The Euclidean distance, denoted as $\|\|$, is used to measure the dissimilarity between a data point x_i and the centroid c_k. The algorithm aims to minimize the WCSS by optimizing the assignment of data points to clusters

4.2.2 Agglomerative Hierarchical Clustering (AGNES)

Agglomerative Hierarchical Clustering (AGNES) is an unsupervised clustering algorithm that can be characterized as "greedy" due to its irreversible steps, as mentioned by Murtagh and Contreras (2012). The algorithm begins by treating each data point as a separate cluster and iteratively merges the two clusters with the smallest distance until a single cluster remains, containing all the data points (Müllner, 2011). For a detailed description of the algorithm, refer to Appendix H.

In R, the default linkage method used in AGNES is average linkage. However, extensive literature has shown that Ward's method performs better than the default method and other alternative methods (Mojena, 1977; Kuiper & Fisher, 1975; Blashfield, 1976). Therefore in our study, we employ Ward's method instead. Ward's method selects the two clusters to merge in each iteration based on minimizing the information loss, which is quantified using the within-cluster sum of squares, as described in Equation 12. This selection process ensures that the clustering results in minimal information loss and maximizes the quality of the clusters (Sharma, Batra et al., 2019).

4.3 Internal Validation of Clustering Quality

We will use four different internal validation indices to evaluate the quality of the clusters. These include the Silhouette Index, Dunn Index, Calinski-Harabasz Index, and Davies-Bouldin Index. The Silhoutte measures the closeness of points in one cluster to the neighbouring clusters (Gupta & Panda, 2019). The Dunn Index quantifies the degree of compactness of clusters and the degree of separation between clusters (Ben Ncir, Hamza & Bouaguel, 2021). The Davies-Bouldin Index looks at the inter- and intra-cluster distance to determine cluster quality. A lower intra-cluster distance is better, as is a higher inter-cluster distance (Mughnyanti, Efendi & Zarlis, 2020). Finally, the Calinski-Harabasz uses a so called variance ratio criterion which is the

ratio between the between-cluster sum of squares and the within-cluster sum of squares. The higher the magnitude of this ratio the better the cluster quality (Caliński & Harabasz, 1974).

5 Results

5.1 Replication

5.1.1 Optimal amount of clusters

In the research conducted by Renjith et al. (2021), the initial step involved determining the optimal number of clusters using the NbClust function. This function provides 26 indices that assess the clustering solutions for different numbers of clusters based on the specified index. However, the paper by Renjith et al. (2021) did not provide specific details about the input parameters, particularly the distance and method used in the NbClust function. Given that the default method is k-means, which is widely utilized in clustering analysis (Charrad, Ghazzali, Boiteau & Niknafs, 2014), and considering that Renjith et al. (2021) employed k-means in their research, it is reasonable to assume that they used k-means as the clustering method. Moreover, the authors did not specify the distance parameter for the NbClust function. Several existing studies (Madhulatha, 2012; Omran, Engelbrecht & Salman, 2007; Kumar, Chhabra & Kumar, 2014; Sinwar & Kaushik, 2014) highlight that the Euclidean distance measure is commonly employed in clustering analysis. Therefore, we can infer that Renjith et al. (2021) may have also utilized the Euclidean distance measure in their study.

In Figure 2, we display the distribution of cluster counts across all indices obtained from the NbClust function. For specific information regarding the preferred number of clusters for each index, please refer to Appendix D, which contains detailed figures.

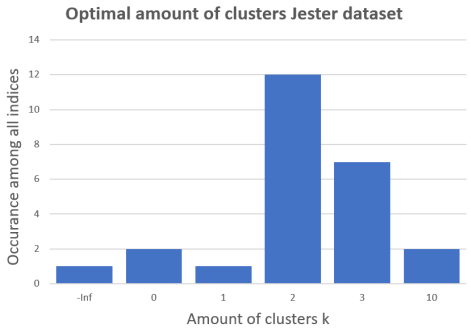


Figure 2: Optimal amount of clusters Jester dataset

The optimal amount of cluster for the Jester dataset as observed from Figure 2 is two which differs from the findings of Renjith et al. (2021), who reported three clusters. To maintain consistency with Renjith et al. (2021), we will conduct the analysis for all datasets using three clusters. However, in Appendix H, we will also perform the analysis using two clusters to explore potential variations in the results.

5.1.2 Dimensionality reduction and clustering

For the PCA and ICA function we need to specify the amount of components to retain. We do this based on scree plots which can be found in Appendix E Figure 13 (see Section 4.1.1 for more details scree plots). Based on this analysis, we conclude that retaining 1 component for PCA and 2 components for ICA is appropriate. The decision is based on the observation of a significant decrease in eigenvalue, which indicates the retention of meaningful information. However, since visualizations with only 1 component are not practical, we choose to retain 2 components for PCA as well.

To assess the assumption of independent and non-Gaussian components in ICA, we examine the correlation between the two components. In this case, the correlation is so negligible that the components can be deemed as uncorrelated. Additionally, we conduct the Shapiro-Wilk test for both independent components and find that the null hypothesis of Gaussianity is rejected at a 5% significance level (refer to Appendix F Table 4 for the detailed results). These results support the assumption of independent and non-Gaussian components for the Jester dataset.

When using the LLE function, it is necessary to specify the number of neighbors. We employ the calc_k() function from the "lle" package to determine the optimal number of neighbors which uses the algorithm by Kayo (see Section Appendix G). Based on the result, we use 51 neighbors when applying the LLE dimensionality reduction method.

After applying the dimensionality reduction techniques, we proceed with conducting k-means clustering and AGNES. In our analysis, we set the number of centers and clusters to three for both methods, following the results obtained from NbClust.

The resulting clusters, along with the clusters obtained without applying any dimensionality reduction, are presented in Figure 3.

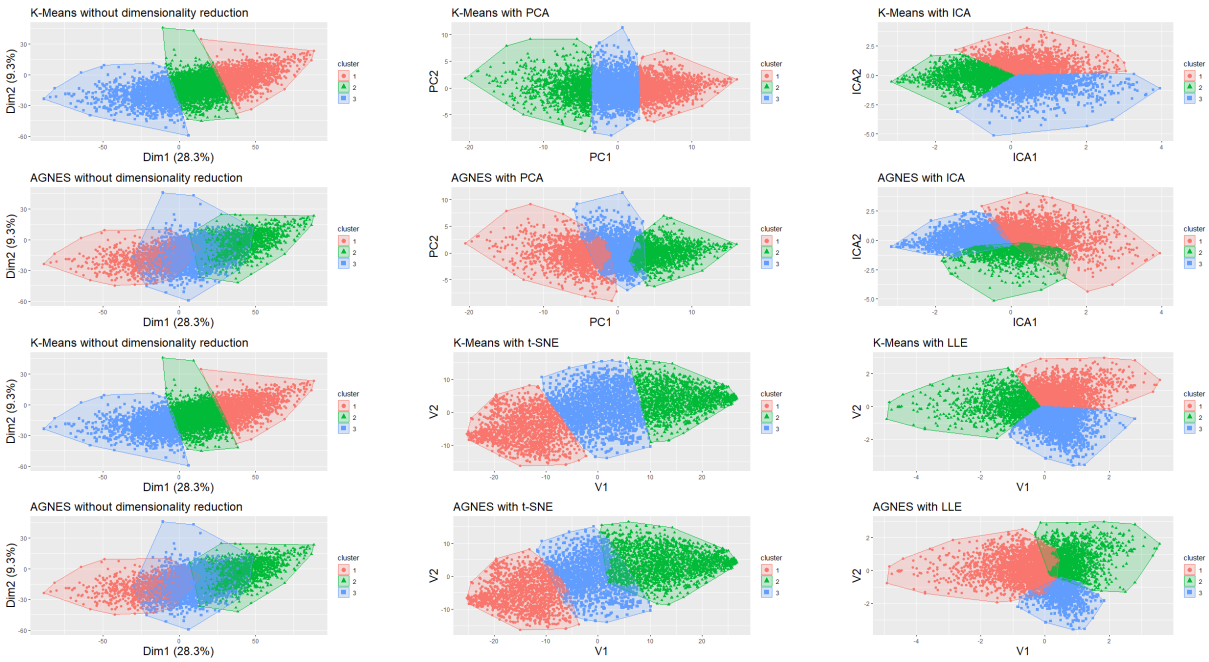


Figure 3: Two-dimensional view of clusters formed with dimensionality reduction techniques when cardinality of the Jester sample is 5000

Based on the observations derived from Figure 3, it is evident that the k-means algorithm generally yields more distinct and separated clusters compared to AGNES, where some clusters show overlap. Additionally, it appears that t-SNE, when combined with both k-means and AGNES, produces the most tightly packed clusters. This suggests that the data points within each cluster are closely situated, indicating a higher quality of clustering.

The characteristics of the clusters obtained in our study exhibit similarities to those reported by Renjith et al. (2021) in terms of separation and compactness. However, we did notice that our obtained clusters display slightly greater overall compactness than those of Renjith et al. (2021), implying that our clusters may possess slightly higher quality.

5.1.3 Cluster quality

Lastly, we measure the quality of the clusters using the four indices: Silhouette Index, Dunn Index, Calinski–Harabasz Index and Davies–Bouldin Index. The results of these indices is shown in Figure 4, 8 and Figure 9.

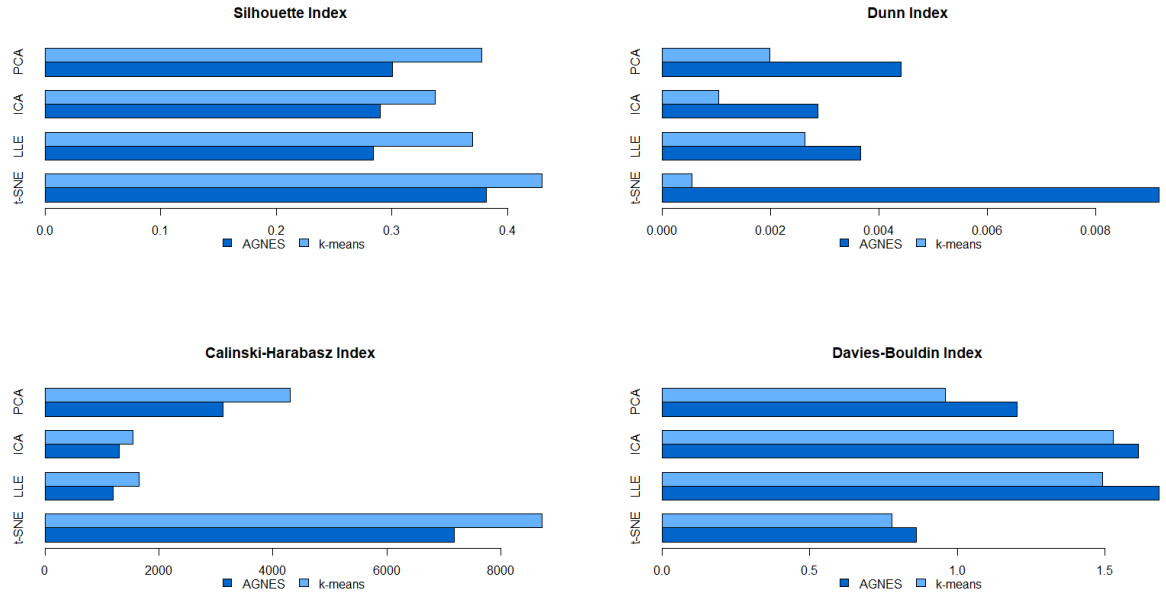


Figure 4: Internal evaluation indices for different dimensionality reduction techniques for the Jester dataset

In Figure 4 we observe the index values for the Jester dataset. The Silhouette Index is a measure used to evaluate the quality of clustering, where higher values indicate better clustering. By analyzing Figure 4, we can see that both k-means and AGNES algorithms have the highest Silhouette Index when used with t-SNE. Moreover, the Dunn Index, which benefits from higher values, identifies t-SNE with AGNES and LLE with k-means as the top-performing methods. Additionally, the Calinski-Harabasz Index also favors t-SNE, indicating that it produces superior clusters for both k-means and AGNES. Lastly, according to the Davies-Bouldin Index, combining t-SNE with both AGNES and k-means results in the lowest value, implying that t-SNE yields better cluster quality.

Considering the consistent out-performance of t-SNE over other dimensionality reduction

techniques across the majority of indices, it is reasonable to conclude that t-SNE demonstrates superiority, which is consistent with the findings of Renjith et al. (2021). This conclusion remains consistent when utilizing two clusters as well. However, there are slight variations in the results when working with two clusters. In this scenario, t-SNE surpasses the other methods across all four indices. Notably, in terms of the Dunn Index, t-SNE now outperforms LLE with k-means. Another disparity between our findings and those of Renjith et al. (2021) is that we observe notably different values for certain dimensionality reduction techniques with respect to some indices. In Renjith et al. (2021), the Dunn Index value was similar for t-SNE with k-means and AGNES, but in our results, t-SNE with k-means performed significantly worse than k-means. Conversely, concerning the Calinski-Harabasz Index, we find that t-SNE with k-means and AGNES exhibit similar performance, whereas Renjith et al. (2021) reported that the value for k-means with t-SNE was nearly three times higher than that of AGNES with t-SNE.

These differences in results observed between our study and the study conducted by Renjith et al. (2021) could be attributed to various factors, such as the use of different sub-samples and potential variations in data cleaning approaches resulting in different clusters. Since the specific data cleaning methods employed by Renjith et al. (2021) were not explicitly stated, it is not possible for us to replicate their exact procedures.

5.2 Extension

5.2.1 Optimal amount of clusters

As explained in detail in Section 5.1.1, we utilize the NbClust function to determine the optimal number of clusters. We employ the Euclidean distance and k-means method as parameter inputs for this analysis.

In Figure 5, we display the distribution of cluster counts across all indices obtained from the NbClust function. For specific information regarding the preferred number of clusters for each index, please refer to Appendix D, which contains detailed figures.

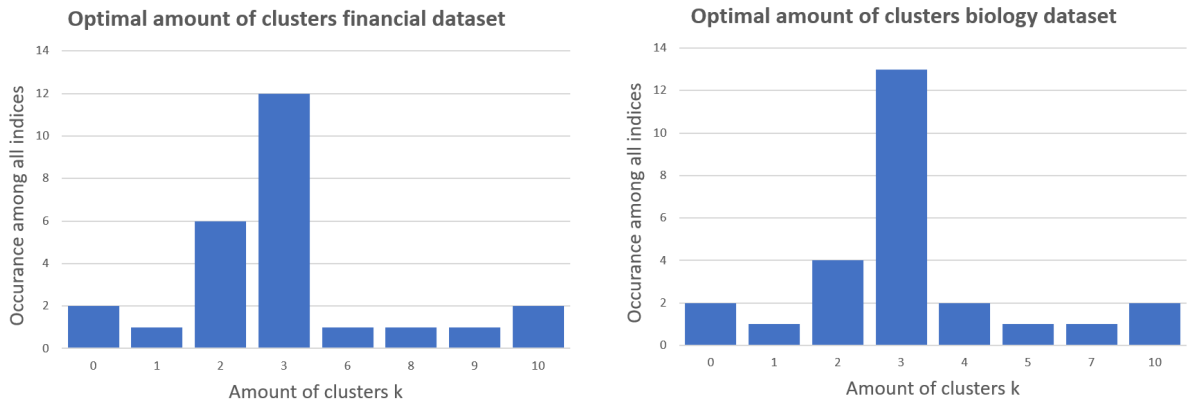


Figure 5: Optimal amount of clusters for the financial dataset (left) and biology dataset (right)

The majority of indices suggest that the optimal number of clusters for both the biology and financial datasets is three. Hence, we will utilize three clusters when applying k-means and AGNES to these datasets.

5.2.2 Dimensionality reduction and clustering

For PCA and ICA function we need to specify the amount of components to retain. We do this based on scree plots which can be seen in Appendix E Figure 14 and 15 (see Section 4.1.1 for more details about scree plots). Based on this analysis, it is suggested to retain 3 components for both PCA and ICA in the financial dataset. For the biology dataset, we choose to retain 2 components for both PCA and ICA based on the scree plot. Evaluating the assumption of independent and non-Gaussian components in ICA, the correlation between the two components for both dataset is so negligible that the components can be deemed as uncorrelated. Additionally, the Shapiro-Wilk test rejects the null hypothesis of Gaussianity for both datasets at a 5% significance level, affirming the validity of the ICA assumption (refer to Appendix F Figure 5 and 6 for the detailed results).

To apply the LLE function, it is necessary to specify the number of neighbors. By employing the calc_k() function from the "lle" package which implements the algorithm by Kayo (see Appendix G), we determine that 20 neighbors are appropriate for both the financial and biology datasets.

After applying the dimensionality reduction techniques, we proceed with conducting k-means clustering and AGNES. In our analysis, we set the number of centers and clusters to three for both methods, following the results obtained from NbClust.

The resulting clusters, along with the clusters obtained without applying any dimensionality reduction, are presented in Figure 6 and 7.

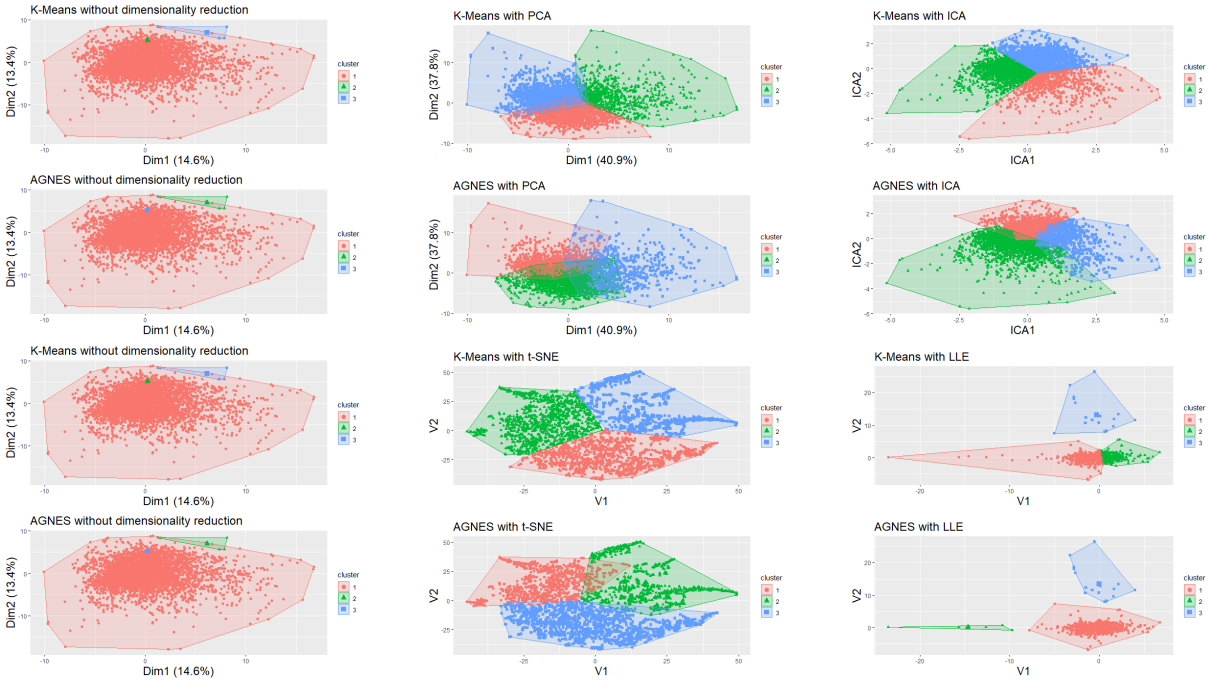


Figure 6: Two-dimensional view of clusters formed with dimensionality reduction techniques when cardinality of the financial sample is 5000

Figure 6 reveals an intriguing observation regarding the clusters obtained without employing dimensionality reduction. Notably, these clusters consist of a large cluster encompassing the majority of observations, alongside two smaller clusters. Conversely, when utilizing LLE, the

resulting clusters appear relatively smaller and more scattered in comparison to clusters obtained using other dimensionality reduction techniques. While the well-separated characteristic of the LLE clusters suggests a favorable quality, the significant difference in density between these clusters raises concerns. This observation is in line with the conclusions drawn by Fernández et al. (2013) in their study, where they anticipated such limitations when using LLE for cluster visualization.

Upon comparing the visual representations of the financial dataset with the Jester dataset, a clear distinction arises in terms of cluster density. Several clusters in the current dataset exhibit a lower density of data points relative to other clusters, indicating a sparser distribution. Specifically, when LLE is applied, the resulting clusters in the current dataset demonstrate noticeable dissimilarities compared to those observed in the Jester dataset. These dissimilarities manifest as smaller, disconnected clusters in the current dataset. Furthermore, it is worth mentioning that the clusters formed without dimensionality reduction appear to be more evenly distributed in the Jester dataset compared to the financial dataset. Lastly, a consistent pattern emerges regarding cluster separation: k-means yields better-separated clusters than AGNES.

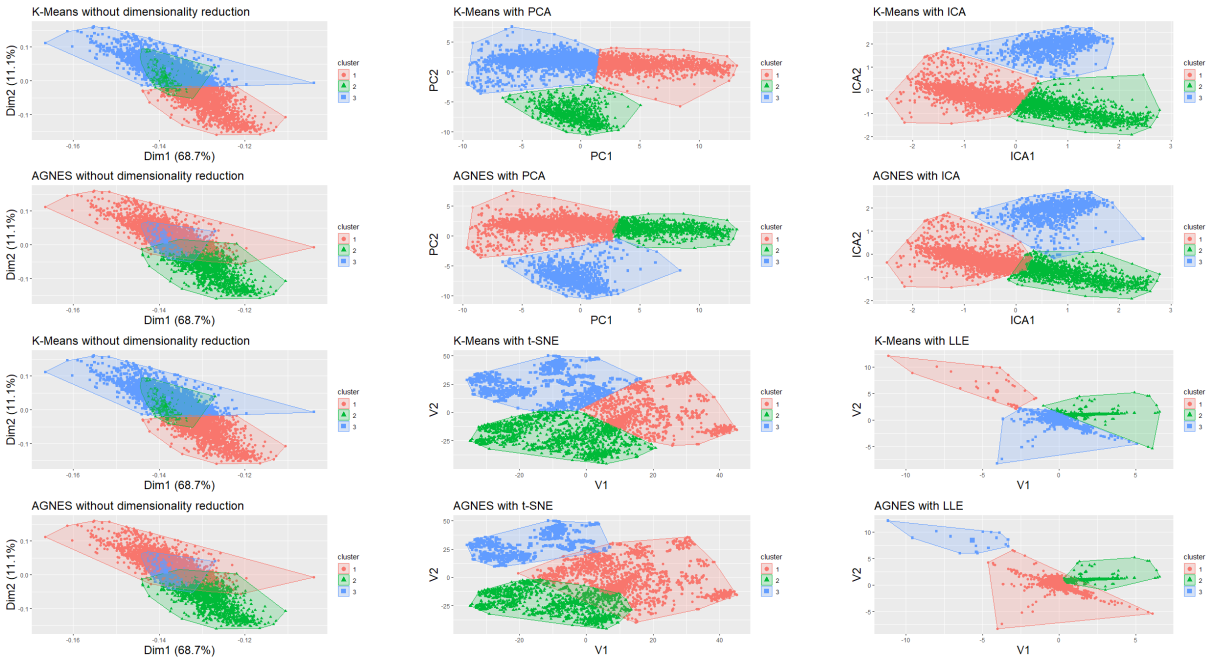


Figure 7: Two-dimensional view of clusters formed with dimensionality reduction techniques when cardinality of the biology sample is 5000

As can be observed from Figure 7 the clusters without any dimensionality reduction exhibit significant overlap, one cluster is even swallowed by the other two, indicating poor cluster quality if no dimensionality reduction is applied. This is a contrast to the Jester and financial dataset. However, when dimensionality reduction techniques are applied, the clusters show reduced overlap. Furthermore, when conducting LLE, it is noticeable that the data points within the clusters exhibit a higher concentration around a central point and are less connected compared to the clusters of the other techniques of dimensionality reduction. In addition, the clusters in the biology dataset exhibit a varying within cluster density of data points than in the Jester dataset and the financial dataset, indicating a sparser distribution for this dataset.

5.2.3 Cluster quality

Lastly, we measure the quality of the clusters using the four indices: Silhouette Index, Dunn Index, Calinski–Harabasz Index and Davies–Bouldin Index. The results of these indices are shown in Figure 8 and Figure 9.

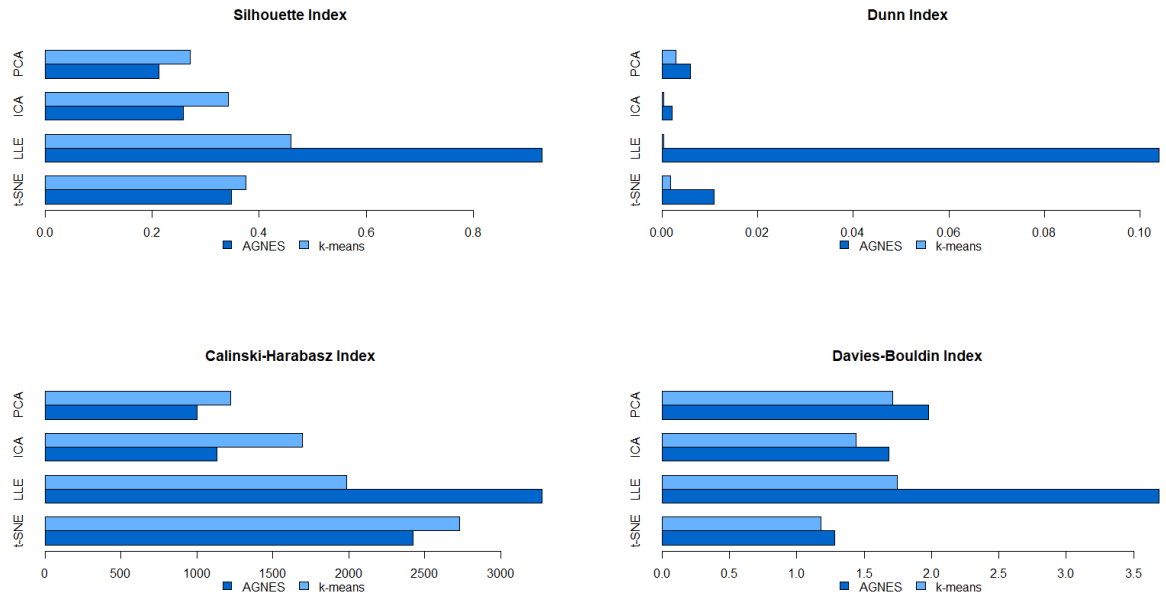


Figure 8: Internal evaluation indices for different dimensionality reduction techniques for the financial dataset

The index results for the financial dataset are presented above. From Figure 8, it is evident that LLE demonstrates the highest values for the Silhouette Index with both AGNES and k-means, indicating its superiority over the other methods. This is also the case for LLE with AGNES concerning the Dunn Index, although PCA outperforms the other methods when paired with k-means. When considering the Calinski-Harabasz Index, t-SNE with k-means and LLE with AGNES yield the best results. The Davies-Bouldin Index shows that the lowest values are achieved by both AGNES and k-means when paired with t-SNE, indicating the superiority of t-SNE in this regard.

Contrary to the Jester dataset, where t-SNE consistently outperforms other methods, the financial dataset reveals LLE as a formidable competitor. In certain scenarios, when paired with specific clustering algorithms and indices, LLE surpasses t-SNE in terms of performance. LLE demonstrates superior preservation of the global structure of the data compared to t-SNE. This can be attributed to the presence of significant dependencies within the financial dataset, which greatly contribute to its overall structure and relationships. LLE's emphasis on preserving the global structure enables it to effectively capture and represent these dependencies in the reduced-dimensional space, surpassing the performance of t-SNE. On the other hand, the Jester dataset consists of lightly dependent variables, explaining why t-SNE performs better for that dataset.

Examining the overall accuracy of all clusters for both the Jester and financial dataset, most indices indicate higher accuracy for the Jester dataset, which is unexpected considering previous findings by Zubova et al. (2018) suggesting that higher dimensionality leads to lower accuracy.

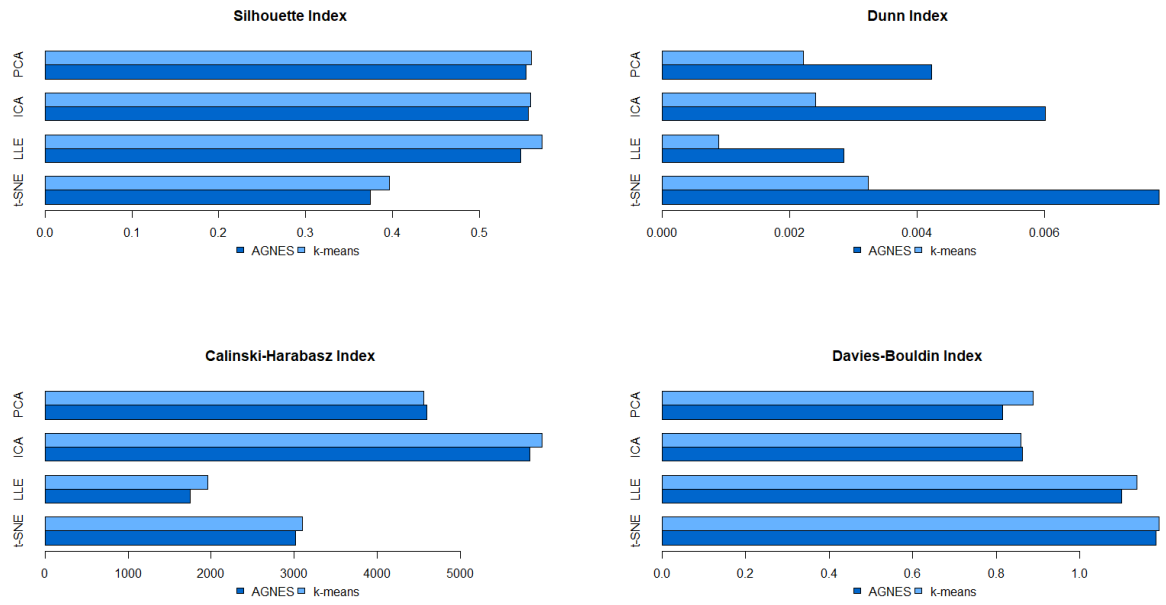


Figure 9: Internal evaluation indices for different dimensionality reduction techniques for the biology dataset

The index results for the biology dataset are presented in Figure 9 above. The Silhouette Index indicates that ICA, PCA, and LLE perform similarly, while t-SNE performs significantly worse than them. On the other hand, the Dunn Index demonstrates the superiority of t-SNE with both AGNES and k-means. In terms of the Calinski-Harabasz Index, ICA outperforms all methods with both AGNES and k-means. Finally, the Davies-Bouldin Index reveals that both PCA and ICA outperform LLE and t-SNE.

In the biology dataset, ICA outperforms t-SNE across various indices, demonstrating its superior performance. This improved performance of ICA can be attributed to its capability to generate independent components that effectively handle the high dependency among variables in this dataset. In contrast, the Jester dataset has less variable dependency, making the production of independent components less crucial in that case. Additionally, when evaluating the overall accuracy of the clusters, the biology dataset, which has fewer variables compared to the Jester dataset, demonstrates better cluster quality, aligning with the findings of Zubova et al. (2018) as expected.

6 Conclusion

The study conducted by Renjith et al. (2021) aimed to examine the performance of different dimensionality reduction methods in clustering analysis using internal clustering validation indices. They evaluated both linear techniques, such as Principal Component Analysis (PCA) and Independent Component Analysis (ICA), and non-linear techniques, including t-Distributed Stochastic Neighbor Embedding (t-SNE) and Locally Linear Embedding (LLE). Their analysis employed k-means clustering and agglomerative hierarchical clustering (AGNES) to assess partitioning and hierarchical clustering.

The findings of Renjith et al. (2021) demonstrated the superior performance of t-SNE when

combined with both k-means and AGNES. In our replication study, we arrived at the same conclusion as Renjith et al. (2021), with the exception that t-SNE combined with AGNES outperformed ICA and LLE in terms of the Davies-Bouldin index, but did not outperform k-means according to the Dunn Index. However, it is important to note that these results may vary when working with datasets of a different nature, as highlighted by Renjith et al. (2021).

To investigate this further, we utilized a biology dataset comprising codon frequencies and a financial dataset consisting of financial ratios of firms. These datasets exhibited distinct dependency patterns among variables, with the financial dataset showing high dependency across certain variables and the biology dataset demonstrating high dependency across nearly all variables. In contrast, the Jester dataset displayed relatively low dependency among variables.

Given the different approaches employed by the dimensionality reduction techniques in handling variable dependencies, we hypothesized that these dissimilarities could impact the results. Indeed, our findings revealed that t-SNE did not exhibit superiority in the financial and biology datasets. Instead, ICA performed better overall for the biology dataset, owing to its ability to generate independent components that effectively address high variable dependency. In the case of the financial dataset, LLE emerged as a strong contender. These findings suggest that when variables exhibit significant inter-dependencies, it becomes crucial for the dimensionality reduction technique to accurately retain or appropriately address these relationships.

The implications of this research are significant, as it underscores the importance of selecting suitable dimensionality reduction methods based on the characteristics of the data. Researchers and organizations can benefit from this knowledge by employing the most appropriate methods for their data analysis, leading to more accurate interpretations and informed decision-making.

Further research can explore alternative dimensionality reduction methods that better account for variable dependencies. One potential approach would be to improve upon the dimensionality reduction techniques used in this study. For instance, incorporating correlation coefficients as a distance metric in LLE, as demonstrated in the paper by Chen and Cao (2012), enhances its performance by explicitly addressing dependencies. This modified version of LLE could be more effective for datasets with strong and numerous variable dependencies. Another potential candidate is partial least squares (PLS), which aims to identify latent variables that explain maximum covariance between the original variables, providing a low-dimensional representation that captures underlying relationships (Rosipal and Krämer, 2005). PLS has the potential to preserve the most important dependencies among variables, making it a promising option for datasets with high dependencies.

References

Abdi, H. & Williams, L. J. (2010). Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, *2*(4), 433–459.

Anowar, F., Sadaoui, S. & Selim, B. (2021). Conceptual and empirical comparison of dimensionality reduction algorithms (pca, kpca, lda, mds, svd, lle, isomap, le, ica, t-sne). *Computer Science Review*, *40*, 100378.

Assent, I. (2012). Clustering high dimensional data. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *2*(4), 340–350.

Athey, J., Alexaki, A., Osipova, E., Rostovtsev, A., Santana-Quintero, L. V., Katneni, U., ... Kimchi-Sarfaty, C. (2017). A new and updated resource for codon usage tables. *BMC Bioinformatics*, *18*, 1–10.

Ben Ncir, C.-E., Hamza, A. & Bouaguel, W. (2021). Parallel and scalable dunn index for the validation of big data clusters. *Parallel Computing*, *102*, 102751. Retrieved from `https://www.sciencedirect.com/science/article/pii/S0167819121000119` doi: https://doi.org/10.1016/j.parco.2021.102751

Blashfield, R. K. (1976). Mixture model tests of cluster analysis: accuracy of four agglomerative hierarchical methods. *Psychological Bulletin*, *83*(3), 377.

Caliński, T. & Harabasz, J. (1974). A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, *3*(1), 1–27.

Charrad, M., Ghazzali, N., Boiteau, V. & Niknafs, A. (2014). NbClust: an R package for determining the relevant number of clusters in a data set. *Journal of Statistical Software*, *61*, 1–36.

De Ridder, D. & Duin, R. P. (2002). Locally linear embedding for classification. *Pattern Recognition Group, Dept. of Imaging Science & Technology, Delft University of Technology, Delft, The Netherlands, Tech. Rep. PH-2002-01*, 1–12.

de Ridder, D., Kouropteva, O., Okun, O., Pietikäinen, M. & Duin, R. P. W. (2003). Supervised locally linear embedding. In *Artificial Neural Networks and Neural Information Processing — ICANN/ICONIP 2003* (pp. 333–341). Berlin, Heidelberg: Springer Berlin Heidelberg.

Ding, C., He, X., Zha, H. & Simon, H. D. (2002). Adaptive dimension reduction for clustering high dimensional data. In *2002 ieee international conference on data mining, 2002. proceedings.* (pp. 147–154).

Dua, D. & Graff, C. (2017). *UCI Machine Learning Repository.* Retrieved from `http://archive.ics.uci.edu/ml`

Fernández, G., Javier, F., Verleysen, M., Lee, J. A. & Ignacio, D. B. (2013). Stability comparison of dimensionality reduction techniques attending to data and parameter variations. In *Eurographics Conference on Visualization (EuroVis) (2013).*

Ge, Z. & Song, Z. (2007). Process Monitoring Based on Independent Component AnalysisPrincipal Component Analysis (ICA-PCA) and Similarity Factors. *Industrial & Engineering Chemistry Research*, *46*(7), 2054-2063. Retrieved from `https://doi.org/10.1021/ie061083g` doi: 10.1021/ie061083g

Goldberg, K., Roeder, T., Gupta, D. & Perkins, C. (2001). Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval*, *4*, 133–151.

Groth, D., Hartmann, S., Klie, S. & Selbig, J. (2013). Principal Components Analysis. In B. Reisfeld & A. N. Mayeno (Eds.), *Computational Toxicology: Volume II* (pp. 527–547). Totowa, NJ: Humana Press. Retrieved from `https://doi.org/10.1007/978-1-62703-059-5`$_2$2 doi: 10.1007/978-1-62703-059-5$_2$2

Gupta, T. & Panda, S. P. (2019). Clustering validation of clara and k-means using silhouette dunn measures on iris dataset. In *2019 international conference on machine learning, big data, cloud and parallel computing (comitcon)* (p. 10-13). doi: 10.1109/COMITCon.2019.8862199

Hartigan, J. A. & Wong, M. A. (1979). Algorithm AS 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society, Series C (Applied Statistics)*, *28*(1), 100–108.

Huang, X., Wu, L. & Ye, Y. (2019). A review on dimensionality reduction techniques. *International Journal of Pattern Recognition and Artificial Intelligence*, *33*(10), 1950017.

James, G., Witten, D., Hastie, T. & Tibshirani, R. (2013). *An Introduction to Statistical Learning: with Applications in R* (2nd ed.). New York, NY: Springer. Retrieved from `https://faculty.marshall.usc.edu/gareth-james/ISL/`

Johnson, R. & Wichern, D. (2014). *Applied Multivariate Statistical Analysis*. Pearson Education Limited.

Kayo, O. (2006). *Locally linear embedding algorithm: extensions and applications.* (Unpublished doctoral dissertation). University of Oulu, Finland.

Khomtchouk, B. B. (2020). Codon usage bias levels predict taxonomic identity and genetic composition. *BioRxiv*, 2020–10.

Kuiper, F. K. & Fisher, L. (1975). 391: A Monte Carlo comparison of six clustering procedures. *Biometrics*, 777–783.

Kumar, V., Chhabra, J. K. & Kumar, D. (2014). Performance evaluation of distance metrics in the clustering algorithms. *INFOCOMP Journal of Computer Science*, *13*(1), 38–52.

Kwon, O. & Sim, J. M. (2013). Effects of data set features on the performances of classification algorithms. *Expert Systems with Applications*, *40*(5), 1847-1857. Retrieved from `https://www.sciencedirect.com/science/article/pii/S0957417412010718` doi: https://doi.org/10.1016/j.eswa.2012.09.017

Madhulatha, T. S. (2012). An overview on clustering methods. *arXiv preprint arXiv:1205.1117*.

Mohamad, I. B. & Usman, D. (2013). Standardization and its effects on K-means clustering algorithm. *Research Journal of Applied Sciences, Engineering and Technology*, *6*(17), 3299–3303.

Mojena, R. (1977). Hierarchical grouping methods and stopping rules: an evaluation. *The Computer Journal*, *20*(4), 359–363.

Mughnyanti, M., Efendi, S. & Zarlis, M. (2020). Analysis of determining centroid clustering x-means algorithm with davies-bouldin index evaluation. In *Iop conference series: Materials science and engineering* (Vol. 725, p. 012128).

Müllner, D. (2011). Modern hierarchical, agglomerative clustering algorithms. *arXiv preprint arXiv:1109.2378*.

Murtagh, F. & Contreras, P. (2012). Algorithms for hierarchical clustering: an overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *2*(1), 86–97.

Na, S., Xumin, L. & Yong, G. (2010). Research on k-means clustering algorithm: An improved k-means clustering algorithm. In *2010 third international symposium on intelligent information technology and security informatics* (pp. 63–67).

Naik, G. R. & Kumar, D. K. (2011). An overview of independent component analysis and its applications. *Informatica*, *35*(1).

National Human Genome Research Institute. (2023). *Codon.* Retrieved from `https://www.genome.gov/genetics-glossary/Codon`.

Oja, E. & Hyvarinen, A. (2000). Independent component analysis: algorithms and applications. *Neural Networks*, *13*(4-5), 411–430.

Omran, M. G., Engelbrecht, A. P. & Salman, A. (2007). An overview of clustering methods. *Intelligent Data Analysis*, *11*(6), 583–605.

Parvathy, S. T., Udayasuriyan, V. & Bhadana, V. (2022). Codon usage bias. *Molecular Biology Reports*, *49*(1), 539–565.

Poole, D. (2015). *Linear algebra: A modern introduction.* CENGAGE Learning.

Razali, N. M., Wah, Y. B. et al. (2011). Power comparisons of Shapiro-Wilk, Kolmogorov-Smirnov, Lilliefors and Anderson-Darling tests. *Journal of Statistical Modeling and Analytics*, *2*(1), 21–33.

Renjith, S., Sreekumar, A. & Jathavedan, M. (2020). Performance evaluation of clustering algorithms for varying cardinality and dimensionality of data sets. *Materials Today: Proceedings*, *27*, 627-633. Retrieved from `https://www.sciencedirect.com/science/article/pii/S2214785320301711` doi: https://doi.org/10.1016/j.matpr.2020.01.110

Renjith, S., Sreekumar, A. & Jathavedan, M. (2021). A comparative analysis of clustering quality based on internal validation indices for dimensionally reduced social media data. *Advances in Artificial Intelligence and Data Engineering: Select Proceedings of AIDE 2019*, *1*, 1047–1065. doi: 10.1007/978-981-15-8864-4$_8$7

Roweis, S. T. & Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, *290*(5500), 2323–2326.

Sambandam, R. (2003). Cluster analysis gets complicated. *Marketing Research*, *15*(1), 16–16.

Shapiro, S. S. & Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika*, *52*(3/4), 591–611.

Sharma, S., Batra, N. et al. (2019). Comparative study of single linkage, complete linkage, and ward method of agglomerative clustering. In *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCON)* (pp. 568–573).

Sinwar, D. & Kaushik, R. (2014). Study of Euclidean and manhattan distance metrics using simple k-means clustering. *Int. J. Res. Appl. Sci. Eng. Technol*, *2*(5), 270–274.

Song, M., Yang, H., Siadat, S. H. & Pechenizkiy, M. (2013). A comparative study of dimensionality reduction techniques to enhance trace clustering performances. *Expert Systems with Applications*, *40*(9), 3722–3737.

Tang, B., Shepherd, M., Milios, E. & Heywood, M. I. (2005). Comparing and combining dimension reduction techniques for efficient text clustering. In *Proceeding of siam international workshop on feature selection for data mining* (pp. 17–26).

Tang, J., Liu, J., Zhang, M. & Mei, Q. (2016). Visualizing large-scale and high-dimensional data. In *Proceedings of the 25th international conference on world wide web* (pp. 287–297).

Tharwat, A. (2021). Independent component analysis: An introduction. *Applied Computing and Informatics*, *17*(2), 222–249.

Van der Maaten, L. & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, *9*(11).

Van Der Maaten, L., Postma, E., Van den Herik, J. et al. (2009). Dimensionality reduction: a comparative. *J Mach Learn Res*, *10*(66-71), 13.

Wharton Research Data Services. (2023). *Financial Ratios Firm Level by WRDS.* Retrieved from `https://wrds-www.wharton.upenn.edu/data-dictionary/wrdsapps`$_{f}$*inratio/*

Zubova, J., Kurasova, O. & Liutvinavičius, M. (2018). Dimensionality reduction methods: The comparison of speed and accuracy. *Information Technology and Control*, *47*(1), 151–160.

A K-means algorithm

Table 2: K-means algorithm pseudo code (Na et al., 2010)

1. Initialize the number of clusters k.
2. Randomly select k data points from the dataset as the initial cluster centroids.
3. Repeat until there is no change in the centroids of the clusters:
    4. Calculate the Euclidean distance between each data point and all cluster centers and assign the data point to the cluster for which this distance is minimized.
    5. Calculate the new centroids of the clusters

B Agglomerative Hierarchical Clustering (AGNES) algorithm

Table 3: AGNES algorithm pseudo code (Murtagh & Contreras, 2012)

1. Assign each data point to it's own cluster. The amount of clusters is now equal to the amount of data points
2. Calculate the distance between each pair of clusters using the Euclidean distance.
3. Repeat until the desired amount of clusters is reached:
    4. Merge the closest clusters based on Ward's criterion: find the pair of clusters with the smallest increase in the within-cluster sum of squares (WCSS) when merged. The increase in WCSS is calculated based on the distance between the merged cluster and the original clusters, as well as the sizes of the clusters involved.
    5. Update the distance matrix: recompute the distances between the merged cluster and all the remaining clusters. Use Ward's criterion to determine the proximity between the merged cluster and other clusters.

C R packages

Application	Fuctions	R package
Principal component analysis (PCA) K-means clustering	prcomp() kmeans()	stats
Independent component analysis (ICA)	preProcess() predict()	caret
t-distributed stochastic neighbour embedding (t-SNE)	Rtsne()	Rtsne
Locally linear embedding (LLE) Optimal number of neighbours to construct datapoint	lle() cal_k()	lle
AGNES clustering Silhouette Index Calinski-Harabasz Index	agnes() silhouette() calinhara()	cluster
Davies-Bouldin Index	index.DB()	clusterSim
Dunn Index	dunn()	clValid
Optimal amount of clusters	NbClust()	clusterCrit
Cluster visualization	fviz_cluster()	factoextra

D Optimal number of clusters for specific indices

Indices	Scott	Cindex	Ball	McClain	Dindex	KL	Silhouette	Duda	PseudoT2	SDBW	Hartigan	Marriot	TrCovW	TraceW	Friedman	Beale	Ratkowsky	CH	CCC	Rubin	DB	Frey	Hubert	PtBiserial	Dunn	Sdindex
K-Means	-Inf	2	3	10	10	2	2	2	2	2	3	5	3	3	3	2	3	2	2	3	2	2	2	1	0	0

Figure 10: Optimal amount of clusters per index for Jester dataset

Indices	Scott	Cindex	Ball	McClain	Dindex	KL	Silhouette	Duda	PseudoT2	SDBW	Hartigan	Marriot	TrCovW	TraceW	Friedman	Beale	Ratkowsky	CH	CCC	Rubin	DB	Frey	Hubert	PtBiserial	Dunn	Sdindex
K-Means	3	2	3	2	0	3	2	2	2	10	3	3	3	3	3	2	10	8	6	3	3	1	0	3	3	9

Figure 11: Optimal amount of clusters per index for financial dataset

Indices	Scott	Cindex	Ball	McClain	Dindex	KL	Silhouette	Duda	PseudoT2	SDBW	Hartigan	Marriot	TrCovW	TraceW	Friedman	Beale	Ratkowsky	CH	CCC	Rubin	DB	Frey	Hubert	PtBiserial	Dunn	Sdindex
K-Means	3	10	3	2	0	7	3	2	2	10	3	3	3	3	3	2	3	3	4	3	3	1	0	4	5	3

Figure 12: Optimal amount of clusters per index for biology dataset

E Scree plots

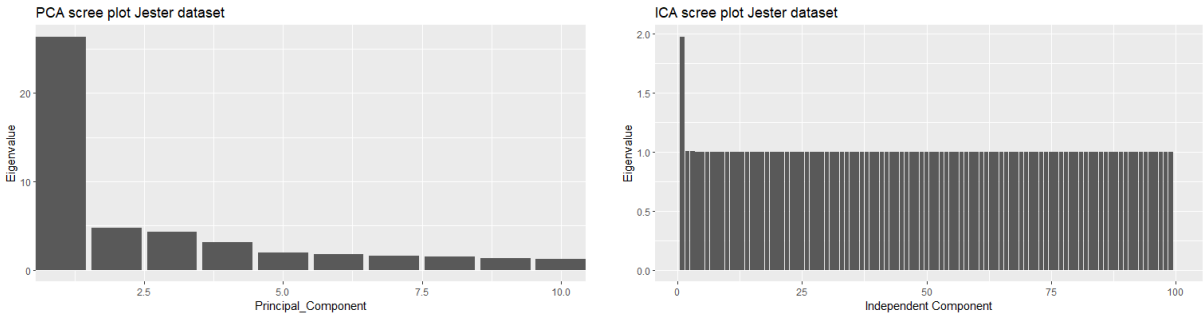


Figure 13: Scree plots for Jester dataset

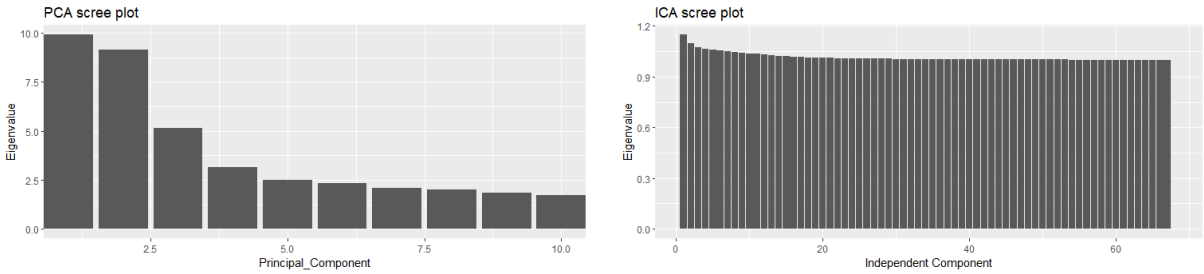


Figure 14: Scree plots for the financial dataset

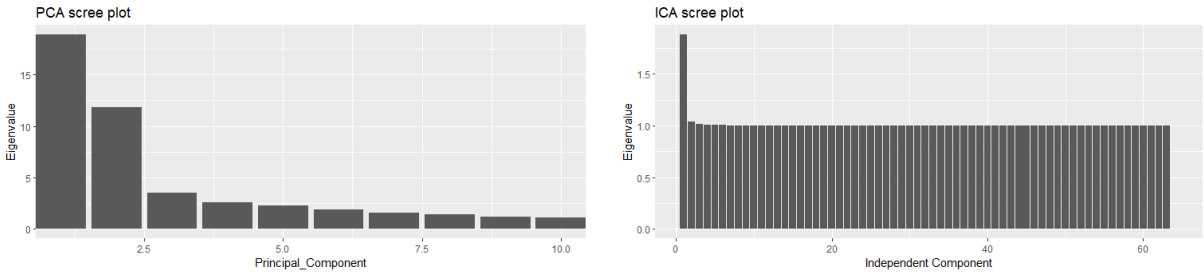


Figure 15: Scree plots for the biology dataset

F Shapiro-Wilk test results

	W	p-value
Independent component 1	0.99803	5.623e-6
Independent component 2	0.99217	6.461e-16

Table 4: Jester dataset Shapiro-Wilk test results

	W	p-value
Independent component 1	0.98393	2.2e-16
Independent component 2	0.96917	2.2e-16

Table 5: Financial dataset Shapiro-Wilk test results

	W	p-value
Independent component 1	0.98346	2.2e-16
Independent component 2	0.96373	2.2e-16

Table 6: Biology dataset Shapiro-Wilk test results

G Optimal number of neighbours algorithm by Kayo

The function calc_k() implements the algorithm proposed by Kayo (2006). This algorithm initially chooses a set of potential candidates for optimal the amount of neighbours, K and an optimality measure is calculated for each candidate. To select the candidates for the optimal number K, the reconstruction error resulting from approximating parts of the nonlinear manifold by linear hyperplanes is considered. This error depends on the weights assigned to data points and the number of nearest neighbors. The function representing the reconstruction error is used as a criterion for identifying potential candidates for the optimal K, which correspond to local and global minima of the function.

After selecting the candidates, the residual variance is computed for each candidate to assess the preservation of distance information. The residual variance measures how well the high-dimensional data is represented in the low-dimensional embedded space. The optimal value of K is determined by selecting the candidate that minimizes the residual variance.

H Replication with two clusters

In order to replicate the analysis with two clusters, we utilize the identical data output for all dimensionality reduction techniques. However, when employing k-means and AGNES, we specifically select two clusters. The resulting clusters can be observed in the accompanying Figure.16.

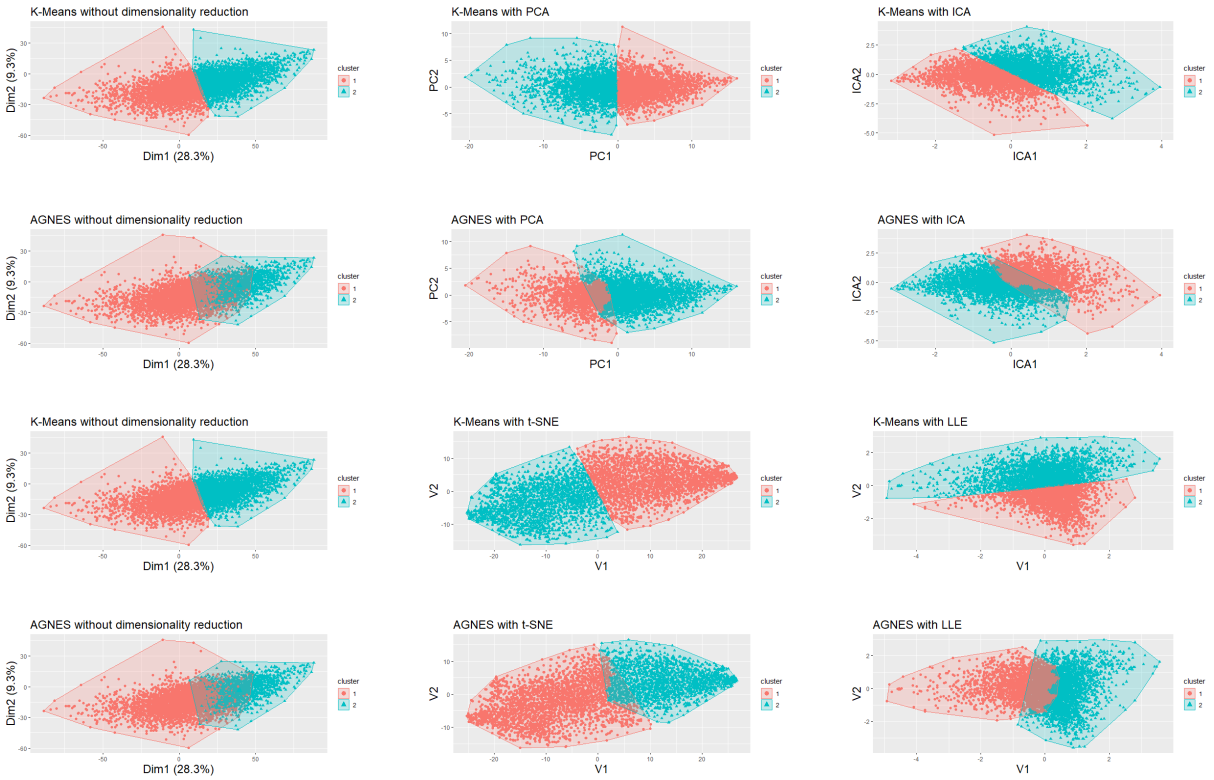


Figure 16: Two-dimensional view of clusters formed with dimensionality reduction techniques when cardinality of the Jester sample is 5000

From Figure 16, it is evident that when using two clusters instead of three, the clusters are evenly distributed in terms of size. However, the level of overlap between the clusters remains consistent across all methods.

Furthermore, we conducted a comprehensive evaluation of cluster quality using four indices: the Silhouette Index, Dunn Index, Calinski-Harabasz Index, and Davies-Bouldin Index. The results of these indices are presented in Figure 17.

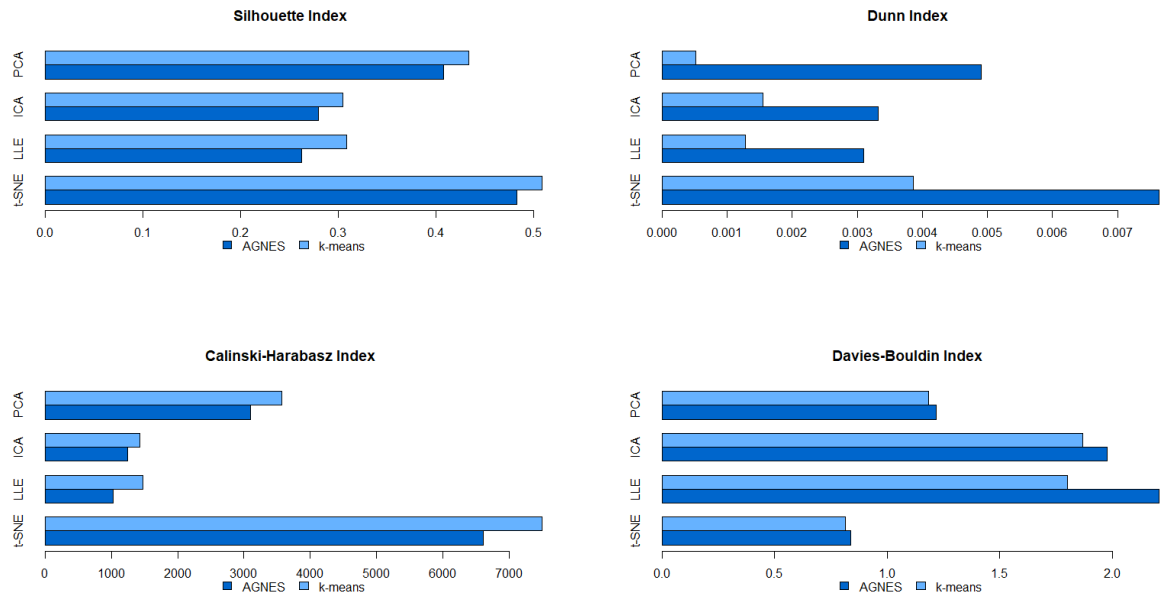


Figure 17: Internal evaluation indices for different dimensionality reduction techniques with the usage of two clusters

Upon examining Figure 17, we can assess the clustering quality using the Silhouette Index, which measures the effectiveness of clustering algorithms. Higher Silhouette Index values indicate better clustering performance. It is evident that both the k-means and AGNES algorithms achieve the highest Silhouette Index when paired with t-SNE. Furthermore, the Dunn Index, which benefits from higher values, identifies t-SNE as the top-performing method. Notably, LLE combined with k-means closely rivals the performance of t-SNE with k-means. The Calinski-Harabasz Index also favors t-SNE, suggesting that it generates superior clusters for both k-means and AGNES. Similarly, the Davies-Bouldin Index indicates that combining t-SNE with both AGNES and k-means yields the lowest value, signifying better cluster quality.

Consistently, when using two clusters, t-SNE maintains its superiority, aligning with the findings of Renjith et al. (2021) in their comparative study. However, there are slight variations in the findings when working with two clusters instead of three. In this scenario, t-SNE outperforms the other methods across all four indices. Notably, in terms of the Dunn Index, t-SNE now surpasses LLE with k-means.