A Proposal for an Ensemble of Clustering Models for High-Dimensional Data (Draft 02)

Antonio Carlos Rodríguez Bajo
Universidad Internacional Menéndez Pelayo
100011543@alumnos.uimp.es

***Abstract*—ADD HERE THE ABSTRACT OF THE PAPER– This thesis must be formatted as a scientific article, including the presentation of the problem to be solved, a review of the state of the art, the proposed solution, the evaluation (theories, results, comparisons) of the solution, and the conclusions. The length of this thesis should be between 6 and 15 pages (IEEE format, double column).**

***Index Terms*—ADD HERE INDEX TERMS**

I. Introduction

CLUSTERING plays an essential role in data analysis as an unsupervised machine learning technique that seeks to group data into meaningful clusters, where elements within the same group share greater similarity compared to those in other groups. This technique has broad applications in diverse fields, including medicine, biology, civil engineering, market research, and social sciences, among others. Despite its utility, it remains a challenging problem due to the absence of ground truth labels, the sensitivity of algorithms to parameter choices, the wide variety of data distributions, and the difficulty of determining the most appropriate number of groups. Moreover, finding optimal clustering solutions is generally an NP-hard problem, which further increases the complexity [1].

A significant theoretical challenge in clustering is presented in Kleinberg's Impossibility Theorem [2]. This theorem asserts that no clustering algorithm can simultaneously satisfy three desirable properties: scale-invariance (the clustering does not change if all distances are multiplied by a positive constant), richness (every possible partition of the data can be obtained by the algorithm for some distance function), and consistency (if distances within clusters are decreased and distances between clusters are increased, the clustering does not change). This result underscores the inherent difficulty of achieving global optimal clustering solutions, particularly when dealing with real-world data that do not adhere to ideal assumptions.

The challenges of clustering are intensified in high-dimensional spaces due to the "curse of dimensionality" [3]. In this scenario, data points tend to become sparse, and the concept of distance, a fundamental aspect of many clustering algorithms, becomes less meaningful [4]. As the number of dimensions increases, the relative distance between the nearest and farthest points diminishes, leading to poor discrimination of distances. This situation not only complicates the identification of meaningful clusters but also increases computational complexity, making traditional clustering algorithms less effective.

Additionally, high-dimensional data often include irrelevant or noisy features that can confound clustering algorithms. For example, algorithms dependent on Euclidean distance may struggle to differentiate between relevant and irrelevant dimensions, further degrading clustering performance [5]. These issues highlight the importance of identifying relevant features and employing advanced preprocessing techniques, such as feature selection and dimensionality reduction, to improve clustering outcomes.

To address these challenges, ensemble clustering has emerged as a promising approach. This method refers to the process of combining multiple clustering results into a single consensus partition, leveraging the strengths of individual algorithms while mitigating their weaknesses. The key idea is to achieve greater robustness, stability, and accuracy by aggregating diverse clustering results [6]. This approach is especially beneficial in high-dimensional settings, where different algorithms may capture complementary aspects of the data [7]. Furthermore, ensemble methods can incorporate feature selection and dimensionality reduction techniques to generate diverse base clusterings and explore various data subspaces, ultimately enhancing the robustness and effectiveness of the final clustering solution [5].

This study introduces an ensemble clustering framework specifically designed for high-dimensional data. The proposed approach integrates unsupervised feature selection, dimensionality reduction techniques and the generation of diverse base clusterings using multiple algorithms and parameter settings. To further enhance robustness, the framework incorporates ensemble selection strategies to retain only the most diverse and high-quality clusterings before applying state-of-the-art consensus functions. By leveraging these components, the framework aims to address the challenges of sparsity, noise, and computational complexity inherent in high-dimensional data, ultimately delivering a robust, scalable, and accurate clustering solution.

ADD HERE A SUMMARY OF THE MAJOR FINDINGS OF THIS PAPER - INCORPORATE A NOVEL CONTRIBUTION NOT CONSIDERED BEFORE (e.g., Systematically benchmark combinations of feature selection, dimensionality reduction, ensemble selection, and consensus functions on a wide range of real and synthetic high-dimensional datasets / Construct hybrid ensembles that simultaneously vary clustering algorithms, feature subsets, and dimensionality reduction method / Others to be explored.

The rest of the paper is structured as follows. Section II provides a comprehensive review of the state of the art, discussing topics such as feature selection, dimensionality reduction techniques, generation of base clusterings, ensemble selection, consensus functions, validation metrics and evaluation. Section III introduces the proposed solution, an ensemble of clustering models designed specifically for high-dimensional data. Section IV presents the experimental results and evaluation, including comparisons with existing methods and analysis of the solution's performance. Finally, Section V concludes the paper by summarizing the findings and outlining potential directions for future research.

II. Review of the State of the Art

The application of ensemble methods in clustering high-dimensional datasets has gained significant attention in recent years due to their potential to improve clustering accuracy, robustness, and stability. These methods leverage multiple clustering solutions to derive a consensus clustering that addresses the inherent challenges of high-dimensional data, such as sparsity, noise, and irrelevant and redundant features.

The concept of cluster ensembles as a knowledge reuse framework, which enables the combination of multiple clustering solutions into a single, robust consensus, was introduced in [6]. A key advantage of this approach is its ability to operate without requiring access to the original data features or the algorithms that produced the initial clusterings; instead, it relies solely on the cluster labels provided by each clustering solution. Three consensus functions—Cluster-based Similarity Partitioning Algorithm (CSPA), HyperGraph Partitioning Algorithm (HGPA), and Meta-CLustering Algorithm (MCLA)—were proposed and demonstrated improved clustering quality and robustness across various datasets, including high-dimensional ones. The use of hypergraph representations facilitates effective integration of multiple clusterings, and the ensemble approach can help mitigate some challenges associated with high-dimensional data.

Probabilistic and statistical approaches for clustering ensembles, including a mixture model of multinomial distributions for consensus clustering, were explored in [8], [9]. Ensemble methods can successfully combine "weak" clusterings—such as those derived from random subspaces, projections, or random hyperplane splits—to produce a superior overall clustering solution. While the mixture model approach requires specifying parameters like the target number of clusters and its performance can be influenced by the diversity and resolution of base clusterings, empirical results show that even simple or weak base clusterings can yield strong consensus results when properly combined.

A bipartite graph partitioning approach for combining cluster ensembles was introduced in [10], demonstrating improved and robust clustering results. To generate diverse base clusterings for high-dimensional data, random projections were used as one ensemble generation method, providing different views of the data and increasing clustering diversity. Although this approach enhances ensemble diversity, both diversity and quality were noted to impact final performance, and highly diverse ensembles (such as those from random projection) may pose challenges for some methods that rely on correspondence between clusters.

Weighted cluster ensemble methods that assign feature-level weights within clusters produced by subspace clustering algorithms were proposed in [7]. This approach improves ensemble performance by leveraging the diversity of clusterings generated with varying parameters and by embedding feature relevance information into the consensus process. The study highlights the challenge of parameter selection in subspace clustering and demonstrates that ensemble methods can provide robust and accurate consensus clusterings without requiring prior knowledge or expensive validation of input parameters.

Recent research has increasingly focused on the scalability of ensemble clustering methods. As discussed in the survey [11], advanced techniques such as graph-based and pairwise similarity-based approaches have been developed to combine clustering results efficiently while preserving solution quality. The survey highlights the strengths of ensemble clustering in integrating heterogeneous data types and providing robust solutions in the face of data perturbations, attributes that have proven particularly valuable in fields like bioinformatics and cybersecurity. However, the authors acknowledge that scalability remains a significant challenge, especially for similarity-based consensus functions, which often have quadratic complexity with respect to the number of data points.

Practical recent applications of ensemble clustering include the study [12], introducing an ensemble clustering method for assessing air quality monitoring networks in Mexico, showcasing the versatility of these methods in environmental science. This approach combined clustering ensembles to evaluate pollution patterns in major metropolitan areas, highlighting their adaptability to real-world datasets. Specifically, the authors applied principal component analysis, hierarchical clustering, and k-means within an ensemble framework to identify similar and redundant monitoring stations in Mexico's three largest cities. This methodology not only improved the robustness of network assessment but also provided actionable insights for optimizing air quality monitoring, underlining the value of ensemble clustering in complex environmental applications.

Ensemble clustering methods have become a promising approach to address the challenges of high dimensionality frequently associated to bioinformatics data. An example is the single-cell RNA sequencing (scRNA-seq) data analysis [13]. In scRNA-seq, identifying cell types and understanding cellular heterogeneity are critical, but the data are often complex, sparse, and noisy, which can undermine the stability and accuracy of individual clustering algorithms. By generating multiple clustering partitions—through varying gene features, cell samples, or clustering algorithms—and integrating them using strategies such as voting or hypergraph-based aggregation, ensemble clustering leverages the complementary strengths of different methods and compensates for their individual weaknesses. This integration yields more robust,

accurate, and stable clustering outcomes, thereby enhancing the identification of cell types and the interpretation of cellular heterogeneity in scRNA-seq datasets.

A. Clustering Ensemble Framework

Clustering ensemble methods seek to combine multiple clustering results into a single, superior partition. Several properties have been proposed as desirable criteria for the effectiveness and reliability of clustering ensemble algorithms [9], [14], [15]:

- Robustness: The ensemble should outperform or, at minimum, match the average performance of individual clustering algorithms, especially in the presence of noise and outliers.
- Consistency: The consensus partition should be similar to the individual clusterings being combined, reflecting the structural agreement among base clusterings.
- Novelty: The ensemble method should be able to discover clustering solutions that are unattainable by any single base clustering algorithm alone, potentially revealing new data structures.
- Stability: The resulting clustering should exhibit low sensitivity to variations in the data, parameter settings, and algorithmic randomness, providing reliable results across runs.

The general process of the clustering ensemble consists of six key steps [11], [16]–[18], as illustrated in Figure 1:

1) Feature selection and preprocessing: The initial step involves identifying and selecting the most relevant features from the high-dimensional dataset, possibly utilizing unsupervised feature selection techniques to reduce noise and irrelevant information. This may also include standard data preprocessing steps such as normalization and handling missing values.
2) Dimensionality reduction: To further mitigate the curse of dimensionality, dimensionality reduction techniques such as Principal Component Analysis (PCA) or random projections methods are applied. These techniques transform the data into lower-dimensional subspaces while preserving as much structural information as possible.
3) Generation of base clusterings: Multiple base clustering solutions are generated by applying different clustering algorithms, or the same algorithm with varying parameters (e.g., number of clusters, initialization, distance metrics), on the original or reduced data. Diversity in base clusterings can also be achieved by subsampling data points or features, or by projecting the data onto random subspaces.
4) Ensemble selection: This is often a beneficial step in the clustering ensemble process, particularly when the pool of base clusterings is large or heterogeneous in quality and diversity. Rather than combining all generated base clusterings, ensemble selection aims to identify and retain a subset of base clusterings that are both high-quality and diverse, thus optimizing the performance of the final consensus solution.

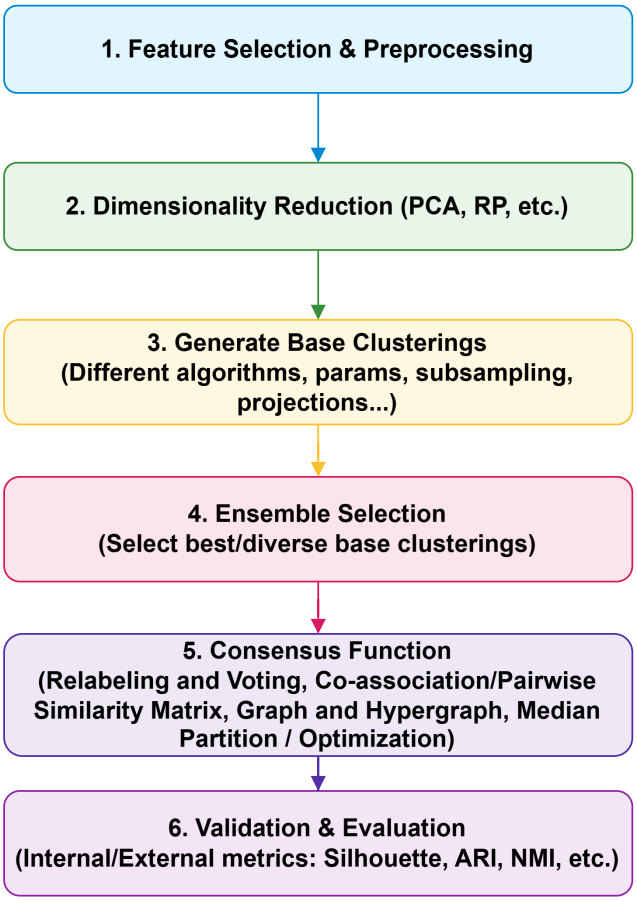

Fig. 1. General process of clustering ensemble

5) Consensus function: The ensemble framework then combines the diverse set of base clusterings into a single consensus partition using a consensus function. Popular approaches include relabeling and voting-based, co-association/pairwise similarity matrix, graph and hypergraph-based, and median partition / optimization-based, each aggregating the input clusterings in different ways to derive a robust final solution.
6) Validation metrics and evaluation: The final clustering solution is evaluated using internal and/or external validation metrics, such as Silhouette Coefficient, Dunn Index, Adjusted Rand Index (ARI), or Normalized Mutual Information (NMI). These metrics assess the quality, stability, and reliability of the consensus clustering, guiding the selection of optimal configurations and ensemble parameters.

The following subsections provide a detailed overview of the key aspects involved in the clustering ensemble process.

B. Unsupervised Feature Selection

Unsupervised feature selection aims to identify the most relevant features of high-dimensional data without relying on labeled examples. It is a critical step in unsupervised learning and clustering tasks, as it improves model interpretability

and reduces computational costs. Unlike supervised feature selection, where class labels guide the search, unsupervised methods must define "interestingness" or "relevance" directly from the data. Here, "interestingness" refers to how well a subset of features uncovers meaningful cluster structure according to a chosen criterion, while "relevance" denotes the contribution of individual features toward revealing these structures in the absence of class labels [19]. Unsupervised feature selection methods can generally be classified into three main categories: filter, wrapper, and embedded approaches, although some methods incorporate hybrid combinations of these primary types.

Filter methods evaluate features based on intrinsic properties of the data, without relying on the outcome of clustering or classification algorithms. They are fast and scalable, making them suitable for large datasets. These methods often use statistical measures or geometric criteria to assess the importance of features. For example, the Laplacian Score [20] evaluates features by their ability to preserve the local manifold structure of the data. This method constructs a nearest neighbor graph to model the local geometric relationships among data points and computes, for each feature, a score that quantifies how well the feature maintains this locality structure. Features with lower Laplacian Scores are considered more relevant for clustering tasks, as they better preserve locality information and are more likely to capture the underlying intrinsic structure of the data.

Wrapper methods integrate feature selection directly with clustering algorithms, evaluating feature subsets based on how well they uncover meaningful cluster structure. Although more computationally intensive than filter or embedded methods, wrappers like Feature Subset Selection using Expectation-Maximization clustering (FSSEM) [19] often yield superior results by systematically searching feature subsets and evaluating each via clustering performance criteria such as scatter separability or maximum likelihood. FSSEM uniquely addresses key challenges in unsupervised feature selection: it determines the optimal number of clusters concurrently with feature selection and corrects for biases related to feature subset dimensionality, making it especially robust for high-dimensional data clustering tasks.

Embedded methods integrate feature selection into the learning model itself, allowing simultaneous optimization of feature selection and the learning of clustering or dimensionality reduction structures. This approach leverages the strengths of machine learning models to identify relevant features during training, making them both efficient and effective for unsupervised tasks such as clustering or dimensionality reduction. An example is the Neural Networks and Self-Expression (NNSE) method [21], which combines neural networks and self-expression models in a unified framework. In the context of feature selection, self-expression refers to modeling each feature (or sample) as a linear combination of all the features in the original space, where the learned weights indicate the importance and redundancy of each feature. By minimizing the reconstruction error, the self-expression model highlights the most representative features while filtering out redundant ones. NNSE replaces traditional linear spectral analysis with neural networks to learn nonlinear relationships between data and pseudo labels, uses self-expression to explore feature relationships and select representative features, and employs adaptive graph regularization to preserve the local manifold structure of the original data.

C. Dimensionality Reduction Techniques

Dimensionality reduction is a critical preprocessing strategy for managing high-dimensional datasets, particularly in unsupervised learning scenarios. It involves transforming features from the original dataset to reduce its dimensionality while retaining as much relevant information as possible [22]. This process helps mitigate issues like the curse of dimensionality, noise, and sparsity, which are common in high-dimensional data. The main techniques applied to the clustering ensemble process are explained below.

Singular Value Decomposition (SVD) factorizes a matrix into three components: left singular vectors, singular values, and right singular vectors. This decomposition enables low-rank approximations of the original matrix, which helps in enhancing pattern detection and highlighting dominant trends in the data [23]. SVD has been effectively utilized as a dimensionality reduction technique before clustering, particularly for handling sparse and high-dimensional datasets. However, SVD is computationally expensive, especially for large datasets, and its sensitivity to outliers and nonlinearities can sometimes degrade clustering performance [22].

Principal Component Analysis (PCA) is one of the most widely used linear dimensionality reduction techniques. PCA applies SVD in a specific way to maximize variance and identify principal components, transforming the original data into a set of orthogonal components called principal components, which capture the maximum variance in the dataset. PCA finds the optimal linear projections using eigenvector decomposition, ensuring minimal information loss [24]. Although PCA is utilized in clustering ensembles to reduce data dimensionality, its effectiveness depends on the dataset and clustering method. PCA combined with random subsampling (PCASS) has been studied for ensemble clustering, where it can improve clustering performance by generating diverse ensemble members, though its success may vary across datasets [17].

Random Projection (RP) is a dimensionality reduction technique that maps high-dimensional data onto a lower-dimensional subspace using a randomly generated projection matrix, while approximately preserving distances between data points. Theoretical guarantees, such as the Johnson-Lindenstrauss lemma [25], ensure that pairwise distances are maintained with minimal distortion. In ensemble clustering, random projections are valuable for generating diverse base clusterings by projecting the data onto different random subspaces, which enhances the robustness and quality of the final consensus clustering. These diverse views help capture various aspects of the data structure, mitigating issues like sparsity and high-dimensional noise [26].

D. Generation of Base Clusterings

The generation of base clusterings involves producing multiple clustering results that can later be combined into a consensus solution. The quality and diversity of the base clusterings are critical for the performance of the ensemble clustering method.

To generate diverse base clusterings, several strategies are commonly employed. One approach involves applying different clustering algorithms, such as k-means, hierarchical clustering, spectral clustering, or density-based clustering, to the same dataset to produce varied results. This method leverages the distinct strengths and criteria of each algorithm, enhancing ensemble diversity [27]. Another strategy focuses on parameter variations, where the same algorithm is run with different configurations, such as varying the number of clusters, initialization methods, or distance metrics. This ensures that even a single algorithm can yield diverse clusterings [11].

Subsampling and random projection offer another effective approach by utilizing random subsets of data points or projecting the data onto random subspaces to create diverse clustering outputs [6], [26], [28]. Additionally, "weak" clustering algorithms, which are simple methods, such as random 1-dimensional projections or random splitting by hyperplanes, that might not perform well individually, are often integrated into ensembles. Their contribution lies in providing diversity, enabling meaningful ensemble results [8], [9].

Ensemble clustering combines a variety of clustering algorithms to leverage their unique strengths and mitigate their individual limitations. Among these, partition-based methods like k-means and k-medoids play a prominent role due to their simplicity and efficiency. These algorithms produce hard partitions of the data, meaning each data point belongs to exactly one cluster. Their computational speed and ease of implementation make them widely used, especially in scenarios where the data conforms to spherical clusters or low-dimensional spaces. However, they may face challenges when applied to high-dimensional or complex data structures [29].

Hierarchical clustering, which constructs dendrograms via single-linkage, complete-linkage, or average-linkage algorithms, is commonly used as one of the base clustering methods in ensemble clustering frameworks. By generating base clusterings with hierarchical algorithms, potentially using different linkage criteria or varying subsets of features or data points, ensemble methods can capture a range of structural patterns and clustering tendencies present in the data. The diversity among base clusterings produced by hierarchical approaches contributes to the robustness and stability of the overall ensemble solution. However, hierarchical clustering can be computationally intensive for large-scale problems, especially when constructing dendrograms for large dataset [11].

Spectral clustering, such as the Ng–Jordan–Weiss method [30], offers a graph-based approach to partitioning data. By leveraging eigenvalue decomposition of similarity matrices, spectral clustering transforms the clustering problem into a graph partitioning challenge. This method is well-suited for non-linearly separable data and excels in detecting clusters with intricate structures. Despite its powerful capabilities, spectral clustering requires careful handling of input parameters and computational resources, particularly for large-scale datasets [27].

Density-based methods such as Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise (DBSCAN) [31] and Ordering Points To Identify the Clustering Structure (OPTICS) [32] focus on identifying clusters based on varying densities. These algorithms are particularly adept at handling data with noise and detecting clusters of arbitrary shapes. Their reliance on density metrics allows them to effectively exclude outliers and uncover patterns in datasets where traditional methods struggle. However, selecting appropriate parameters for density thresholds can be challenging and requires domain expertise [11].

Fuzzy clustering methods, such as Fuzzy C-Means [33], provide a unique approach by allowing data points to belong to multiple clusters with varying degrees of membership. This is particularly useful for ambiguous or uncertain datasets where hard partitions may oversimplify the underlying structure. Fuzzy clustering is effective in capturing nuanced relationships between data points but can be computationally demanding, especially when dealing with high-dimensional data [11], [34].

These diverse clustering algorithms, each with its strengths and weaknesses, can be integrated into the ensemble clustering process to enhance robustness, accuracy, and stability across varying data types and structures.

E. Ensemble Selection

Ensemble selection in clustering ensembles focuses on choosing the best subset of base clusterings to improve consensus results, since including all partitions, especially low-quality or redundant ones, can reduce performance. Selecting clusterings that are both diverse and of high quality can significantly improve the resulting consensus partitions [18].

High quality refers to base clusterings that are internally coherent and capture meaningful structure within the data; in practice, this is often assessed using internal validity measures or by evaluating how well a clustering agrees with the overall trend of the ensemble, such as through the Sum of Normalized Mutual Information (SNMI). Diversity, on the other hand, measures how different the selected clusterings are from each other, ensuring that the ensemble combines complementary perspectives rather than redundant solutions. This is commonly quantified by calculating low pairwise similarity (e.g., low NMI) between clusterings [35].

Recent research has advanced clustering ensemble selection by introducing sophisticated strategies such as hierarchical and multiplex network-based approaches, which utilize ensemble diversity and quality through various validity indices and community detection techniques [36], [37]. Other works have developed selection strategies that explicitly combine internal and external validity measures to guide the process and improve consensus performance [38]. There is also a

growing emphasis on adaptive, data-driven criteria for ensemble selection, where the contribution of each candidate partition is evaluated relative to the ensemble in a dynamic and context-aware manner [35]. Furthermore, recent studies highlight the benefits of ensemble selection for computational efficiency, particularly in large-scale and high-dimensional settings, where reducing ensemble size while maintaining accuracy is essential [11].

F. Consensus Functions

A core component of clustering ensemble methods is the consensus function—the strategy that combines multiple base clusterings into a single, robust consensus partition. The choice of consensus function significantly influences the quality, stability, and interpretability of the final clustering solution. Various consensus functions have been developed, which can be broadly categorized based on the type of information they utilize and their computational mechanism.

In the context of clustering ensembles, n denotes the number of data points or objects being clustered, r (also sometimes denoted as m or H) is the number of base clusterings (partitions) in the ensemble, and k is the number of clusters in the final consensus partition. Below, a taxonomy of consensus functions is presented [6], [9], [11], [15].

1) Relabeling and Voting-Based Approaches: These methods address the label correspondence problem by aligning cluster labels across partitions—typically using the Hungarian algorithm [39], which has a computational cost of $O(k^3)$—before assigning each object to the cluster label that receives the most votes (via plurality or majority voting). Such approaches are most effective for small datasets and for ensembles where the number of clusters is fixed across all partitions, allowing reliable and efficient label alignment. However, as the number or diversity of clusters increases, or when base clusterings contain different numbers of clusters, these methods become less practical due to the ill-posed nature and computational complexity of label alignment. In these cases, the correspondence problem becomes increasingly difficult or ambiguous, and alternative consensus mechanisms may be preferable [40], [41].

Plurality voting assigns each data point to the cluster label with the highest vote after label alignment, and is primarily feasible when all partitions have the same number of clusters [41]. Cumulative voting [40] extends this idea by incrementally updating a reference partition and can accommodate varying cluster numbers using probabilistic mappings. Nevertheless, voting-based approaches are generally not recommended when cluster numbers differ significantly between partitions due to the label correspondence problem and the computational cost of alignment.

2) Co-association/Pairwise Similarity Matrix Methods: These methods construct an $n \times n$ co-association matrix, where each entry (i, j) records the frequency (or proportion) with which objects i and j appear in the same cluster across ensemble members. This matrix serves as a new similarity measure, to which a standard clustering algorithm (such as hierarchical agglomerative clustering) is applied to extract the consensus clustering. Evidence Accumulation Clustering is a prominent example, using the co-association matrix as input for linkage-based clustering to produce the consensus partition. While intuitive and effective for small to medium datasets, this approach has $O(n^2)$ computational and memory complexity, making it unsuitable for very large data sets [14]

3) Graph and Hypergraph-Based Methods: These approaches represent relationships among data points and clusters as graphs or hypergraphs: nodes correspond to data points (or clusters), and edges or hyperedges encode co-membership or similarity. Consensus clustering is achieved by partitioning this (hyper)graph using graph partitioning algorithms.

The Cluster-based Similarity Partitioning Algorithm (CSPA) constructs a similarity (co-association) matrix and interprets it as a weighted adjacency matrix, using a graph partitioning algorithm to derive the consensus. While effective, it suffers from $O(n^2)$ complexity. The HyperGraph Partitioning Algorithm (HGPA) treats each cluster as a hyperedge; the consensus is found by partitioning the hypergraph to cut as few hyperedges as possible, which is computationally more efficient ($O(nkr)$) and well-suited to balanced clusters, but may perform poorly with highly imbalanced clusters. The Meta-CLustering Algorithm (MCLA) constructs a meta-graph of clusters (across partitions), partitions this meta-graph, and then assigns data points to the resulting meta-clusters, often yielding robust consensus results especially when meaningful cluster correspondences exist. These methods scale well to large ensembles and can handle missing data, but CSPA is limited by quadratic complexity and HGPA may be biased toward balanced cluster sizes [6].

4) Median Partition / Optimization-Based Approaches: These methods formulate consensus clustering as an optimization problem: the aim is to find the partition (the "median partition") that maximizes overall similarity (or minimizes dissimilarity) to all base clusterings according to a chosen similarity or distance measure (such as normalized mutual information (NMI), Mirkin distance, or variation of information). The median partition problem is NP-hard, so practical implementations rely on heuristics or approximations (e.g., greedy search, simulated annealing, or k-means in a transformed feature space) [11].

Non-Negative Matrix Factorization (NMF)-based consensus methods encode the ensemble into a matrix and factorize it, yielding soft assignments and efficient computation for larger datasets. These approaches are theoretically well-founded and directly target the consensus objective, but computational costs can be significant for large n or complex similarity metrics [11].

G. Validation Metrics and Evaluation

Evaluating the quality and effectiveness of clustering ensembles is a fundamental yet challenging task, especially in the absence of ground truth labels, a common scenario in unsupervised learning. The assessment of clustering solutions typically relies on cluster validity indices, which can

be broadly categorized into internal, external, and relative indices. Each of these metrics evaluates different aspects of clustering performance and offers complementary insights into the quality of the partitions produced by ensemble methods [11].

Internal validity indices are essential tools for evaluating the quality of clustering solutions without relying on external information such as class labels. These indices assess clustering results by examining the inherent structure of the data, focusing primarily on two key aspects: compactness (how closely related the data points within the same cluster are) and separation (how distinct different clusters are from each other).

Some of the most widely used internal validity indices are the Davies-Bouldin Index, which evaluates the average similarity between each cluster and its most similar one, rewarding compact and well-separated clusters with lower values; the Silhouette Coefficient, which combines measures of both intra-cluster cohesion and inter-cluster separation, with higher average values indicating better clustering; and the Calinski-Harabasz Index and Dunn Index, which rely on ratios of between-cluster and within-cluster distances to quantify cluster definition, each with its own sensitivity to cluster shapes and densities. Other metrics, like the C Index, Gamma Index, PBM Index, Ray-Turi Index, and McClain-Rao Index, introduce alternative formulations for assessing the internal structure of clusters, often considering variances, pairwise distances, or cluster compactness [35].

External validity indices compare clustering results to an external reference or ground truth (when available), providing a direct measure of agreement between the obtained clusters and known class labels. A wide range of external indices have been developed, each capturing different aspects of similarity or agreement. Commonly used indices include the Adjusted Rand Index (ARI) and the Rand Index (RI), which are pair-counting measures that quantify the proportion of sample pairs on which two partitions agree; the Normalized Mutual Information (NMI), Mutual Information (MI), and Adjusted Mutual Information (AMI), which are information-theoretic measures evaluating the shared information between cluster assignments and ground truth; and the Variation of Information (VI) and Entropy, which quantify the information lost or gained between the two partitions.

Set-matching measures such as Purity and F-Measure (F1-score) assess the extent to which clusters contain a single class or the harmonic mean of precision and recall, respectively. Additional pairwise indices like the Jaccard Index and the Fowlkes–Mallows Index also provide valuable perspectives on clustering agreement. More recently, the Chi Index, based on the chi-squared statistical test, has been proposed to directly assess the dependence between clustering and ground truth assignments using contingency tables. [35], [42]

Relative validity indices are designed to compare the quality of different clustering results or parameter settings on the same dataset by quantifying how changes in the clustering structure affect certain quality measures. Unlike absolute indices, which provide an overall assessment of a single clustering, relative indices are particularly useful for evaluating the impact of algorithmic choices, such as the number of clusters, distance metrics, or initialization methods. [35]

Empirical validation in clustering ensemble research frequently relies on combining multiple validity indices to achieve a comprehensive evaluation, particularly for high-dimensional datasets where individual metrics may be inadequate. Benchmarking on both synthetic and real-world datasets, in conjunction with the use of diverse validity indices, is recommended to assess the reliability and generalization of ensemble clustering methods.

III. Solution Proposal

This solution employs ensemble clustering tailored for high-dimensional datasets to enhance robustness, stability, and accuracy. By integrating multiple base clustering results into a unified consensus partition, This approach leverages the strengths of diverse clustering methods, effectively addressing challenges such as sparsity, noise, and the absence of an optimal individual clustering method.

Fundamentally, the method relies on two key principles: diversity in base clusterings, achieved by varying algorithms, parameters, and feature subsets, and a consensus function, responsible for synthesizing these heterogeneous partitions into a single, cohesive clustering solution.

The ensemble strategy proposed here consists of the following steps:

1) Feature Selection & Dimensionality Reduction: Apply unsupervised feature selection and dimensionality reduction to mitigate the curse of dimensionality and enhance the quality of clustering inputs.
2) Generation of Base Clusterings: Generate a diverse set of base clusterings by combining different clustering algorithms, multiple parameter settings, and random subsampling of data points or features.
3) Ensemble Selection: Evaluate and select a subset of high-quality and diverse base clusterings using internal validity indices and diversity measures, to avoid redundancy and poor-performing models.
4) Apply robust consensus functions to aggregate the selected base clusterings into a final, consensus partition.
5) Evaluation: Assess the quality of the ensemble clustering using internal and external cluster validity metrics, benchmarking against single-algorithm baseline or other ensemble approaches.

A. Experimental Framework

The experimental framework is designed to ensure reproducibility, scalability, and extensibility across different computational environments and programming languages.

1) Experimental Framework: TO BE UPDATED Experiments were conducted using a combination of local computing resources (e.g., multicore CPUs, workstation GPUs) and cloud-based virtual machines, as required by dataset size and computational demand. For large-scale or parallel processing,

Google was leveraged to provision powerful CPU and GPU instances, facilitating efficient handling of high-dimensional datasets and computationally intensive ensemble methods.

2) Programming Tools and Libraries: The diceR package [43], [44] in R served as the primary tool for implementing the ensemble clustering framework. This package provides a comprehensive suite of functions for performing cluster analysis using an ensemble clustering approach. It supports the generation of diverse base clusterings through multiple algorithms, the application of various consensus functions, and the evaluation of clustering performance via both internal and external validity indices. To extend its capabilities, an enhanced version named diceRplus [45] was developed, introducing unsupervised feature selection, dimensionality reduction, and additional functionalities.

Auto-UFSTool [46] was utilized for unsupervised feature selection. This MATLAB toolbox provides a collection of 25 robust unsupervised feature selection approaches, most of which were developed within the last years. It enables the evaluation of feature selection results and generates comparative graphs for different feature subsets. By incorporating Auto-UFSTool into the workflow, the ensemble clustering framework can benefit from state-of-the-art unsupervised feature selection techniques, potentially improving the quality and relevance of the input features for subsequent clustering steps.

R.matlab [47] was used to connect R and MATLAB. It is an R package that allows communication between R and MATLAB via a TCP/IP connection. The package provides methods for reading and writing MAT files, as well as sending commands and data between R and MATLAB.

MORE TOOLS/PACKAGES TO BE ADDED AS NEEDED

3) Experimental Design and Reproducibility: To guarantee reproducibility and systematic analysis, an experiment management framework will be established. This included:

- Configuration Management: All experiment configurations (algorithm choices, parameter settings, random seeds, data splits) will be stored in version-controlled files.
- Result Storage: Outputs—including cluster assignments, consensus matrices, and validity scores—will be stored in structured formats for traceability and post-hoc analysis.
- Automation & Analytics: Scripts will be developed to automate the running of experiments, aggregation of results, and generation of analytical visualizations.
- Documentation and Code Sharing: All scripts and documentation will be maintained in a git repository to facilitate transparency.

IV. Experimental Results and Evaluation

This section will include:

- Experimental evaluation: Conduct experiments to evaluate the proposed method using benchmark datasets.
- Performance comparison: Compare the performance of the proposed approach against other ensemble clustering methods found in the state-of-the-art review.

V. Conclusions

This section will cover:

- Summary of key findings: Present the main discoveries of the research, demonstrating how ensemble clustering methods can effectively address the challenges posed by high-dimensional datasets, including feature relevance, data sparsity, and noise.
- Discussion of advantages: Discuss the benefits of the proposed ensemble clustering solution, such as improved robustness, accuracy, and the ability to balance the diversity and quality of base clusterings.
- Directions for future work: Suggest possible future work, such as exploring more advanced consensus functions, integrating additional feature selection and dimension reduction techniques, or applying the framework to other types of data and domains.

References

[1] R. Scitovski, K. Sabo, F. Martínez-Álvarez, and Š. Ungar, *Cluster Analysis and Applications*. Springer, 2021.

[2] J. Kleinberg, "An Impossibility Theorem for Clustering," in *Advances in Neural Information Processing Systems (NIPS)*, vol. 15, 2002.

[3] R. Bellman, *Adaptive Control Processes: A Guided Tour*, Princeton University Press, 1961.

[4] K. Beyer, J. Goldstein, R. Ramakrishnan, and U. Shaft, "When Is Nearest Neighbors Meaningful?" in *International Conference on Database Theory*, pp. 217–235, 1999.

[5] X. Z. Fern and C. E. Brodley, "Random Projection for High Dimensional Data Clustering: A Cluster Ensemble Approach," in *Proceedings of the 20th International Conference on Machine Learning (ICML)*, 2003, pp. 186–193.

[6] A. Strehl and J. Ghosh, "Cluster Ensembles – A Knowledge Reuse Framework for Combining Multiple Partitions," *Journal of Machine Learning Research*, vol. 3, pp. 583–617, Dec. 2002.

[7] C. Domeniconi and M. Al-Razgan, "Weighted Cluster Ensembles: Methods and Analysis," *ACM Transactions on Knowledge Discovery from Data*, vol. 2, no. 4, pp. 1–40, 2009.

[8] A. Topchy, A. K. Jain, and W. Punch, "Combining multiple weak clusterings," in *Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM)*, Melbourne, FL, USA, 2003, pp. 331–338.

[9] A. Topchy, A. K. Jain, and W. Punch, "Clustering ensembles: Models of consensus and weak partitions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 12, pp. 1866–1881, 2005.

[10] X. Z. Fern and C. E. Brodley, "Solving cluster ensemble problems by bipartite graph partitioning," in *Proceedings of the 21st International Conference on Machine Learning (ICML)*, Banff, Alberta, Canada, 2004, pp. 36.

[11] T. Boongoen and N. Iam-On, "Cluster ensembles: A survey of approaches with recent extensions and applications," *Computer Science Review*, vol. 28, pp. 1–25, 2018.

[12] T. Stolz, M. E. Huertas, and A. Mendoza, "Assessment of air quality monitoring networks using an ensemble clustering method in the three major metropolitan areas of Mexico," *Atmospheric Pollution Research*, vol. 11, no. 8, pp. 1271-1280, 2020.

[13] X. Nie, D. Qin, X. Zhou, H. Duo, Y. Hao, B. Li, and G. Liang, "Clustering ensemble in scRNA-seq data analysis: Methods, applications and challenges," *Computers in Biology and Medicine*, vol. 159, p. 106939, 2023.

[14] A. L. N. Fred and A. K. Jain, "Combining multiple clustering using evidence accumulation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 6, pp. 835–850, 2005.

[15] S. Vega-Pons and J. Ruiz-Shulcloper, "A survey of clustering ensemble algorithms," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 25, no. 3, pp. 337–372, 2011.

[16] M. S. Pavithra and R. Parvathi, "Cluster Ensemble Approach for High Dimensional Data," *Australian Journal of Basic and Applied Sciences*, vol. 12, no. 1, pp. 45-53, Jan. 2018.

[17] X. Z. Fern and C. E. Brodley, "Cluster Ensembles for High Dimensional Clustering: An Empirical Study," School of Electrical and Computer Engineering, Purdue University, W. Lafayette, IN, and Electrical Engineering and Computer Science, Tufts University, Medford, MA, 2003.

[18] X. Z. Fern and W. Lin, "Cluster ensemble selection," *Statistical Analysis and Data Mining*, vol. 1, no. 3, pp. 128-141, 2008.

[19] J. G. Dy and C. E. Brodley, "Feature selection for unsupervised learning," *Journal of Machine Learning Research*, vol. 5, pp. 845–889, Aug. 2004.

[20] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," in *Advances in Neural Information Processing Systems*, vol. 18, 2005.

[21] M. You, A. Yuan, D. He, and X. Li, "Unsupervised feature selection via neural networks and self-expression with adaptive graph constraint," *Pattern Recognition*, vol. 135, p. 109173, Mar. 2023.

[22] S. Ayesha, M. K. Hanif, and R. Talib, "Overview and comparative study of dimensionality reduction techniques for high dimensional data," *Information Fusion*, vol. 59, pp. 44–58, 2020.

[23] C. Eckart and G. Young, "The approximation of one matrix by another of lower rank," *Psychometrika*, vol. 1, no. 3, pp. 211–218, Sep. 1936.

[24] Hotelling, Harold. "Analysis of a Complex of Statistical Variables into Principal Components." *The Journal of Educational Psychology*, vol. 24, no. 6, 1933, pp. 417–441.

[25] W. B. Johnson and J. Lindenstrauss, "Extensions of Lipschitz mappings into a Hilbert space," *Contemporary Mathematics*, vol. 26, pp. 189–206, 1984.

[26] L. Anderlucci, F. Fortunato, and A. Montanari, "High-Dimensional Clustering via Random Projections," *Journal of Classification*, vol. 39, no. 2, pp. 191–216, 2022.

[27] S. Vega-Pons and J. Ruiz-Shulcloper, "Clustering Ensemble Method for Heterogeneous Partitions," in *CIARP*, 2009, pp. 481-488.

[28] S. Dudoit and J. Fridlyand, "Bagging to improve the accuracy of a clustering procedure," Bioinformatics, vol. 19, no. 9, pp. 1090–1099, 2003.

[29] Z.H. Zhou, *Ensemble Methods: Foundations and Algorithms*, CRC Press, 2012.

[30] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in Neural Information Processing Systems 14*, T. G. Dietterich, S. Becker, and Z. Ghahramani, Eds. MIT Press, 2002, pp. 849–856.

[31] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD)*, 1996, pp. 226–231.

[32] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, "OPTICS: Ordering points to identify the clustering structure," in *Proceedings of the 1999 ACM SIGMOD International Conference on Management of Data*, 1999, pp. 49–60.

[33] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York: Plenum Press, 1981.

[34] V. Berikov, "Weighted ensemble of algorithms for complex data clustering," *Pattern Recognition Letters*, vol. 38, pp. 99-106, 2014.

[35] K. Golalipour, E. Akbari, S. S. Hamidi, M. Lee, and R. Enayatifar, "From clustering to clustering ensemble selection: A review," *Engineering Applications of Artificial Intelligence*, vol. 104, p. 104388, 2021.

[36] M. C. Naldi, A. C.P.L. de Carvalho, and R. J.G.B. Campello, "Cluster ensemble selection based on relative validity indexes," *Data Mining and Knowledge Discovery*, vol. 27, no. 2, pp. 259–289, 2013.

[37] E. Akbari, H. M. Dahlan, R. Ibrahim, and H. Alizadeh, "Hierarchical cluster ensemble selection," *Engineering Applications of Artificial Intelligence*, vol. 39, pp. 146–156, 2015.

[38] S. O. Abbasi, S. Nejatian, H. Parvin, V. Rezaie, and K. Bagherifard, "Clustering ensemble selection considering quality and diversity," *Artificial Intelligence Review*, vol. 52, no. 2, pp. 1311–1340, 2019.

[39] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, pp. 83–97, 1955.

[40] H. G. Ayad and M. S. Kamel, "Cumulative voting consensus method for partitions with a variable number of clusters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 1, pp. 160–173, 2008.

[41] A. P. Topchy, M. H. C. Law, A. K. Jain, and A. L. N. Fred, "Analysis of consensus partition in cluster ensemble," in *Proceedings of the Fourth IEEE International Conference on Data Mining (ICDM)*, 2004, pp. 225–232.

[42] J. M. Luna-Romera, M. Martínez-Ballesteros, J. García-Gutiérrez, and J. C. Riquelme, "External clustering validity index based on chi-squared statistical test," *Information Sciences*, vol. 487, pp. 1–17, 2019.

[43] D. S. Chiu and A. Talhouk, "diceR: an R package for class discovery using an ensemble driven approach," BMC Bioinformatics, vol. 19, no. 1, p. 11, Feb. 2018.

[44] D. Chiu, A. Talhouk, and J. Liu, "Package 'diceR'," CRAN, Feb. 2025. [Online]. Available: https://CRAN.R-project.org/package=diceR

[45] A. C. Rodriguez Bajo, "diceRplus," GitHub repository, 2025. [Online]. Available: https://github.com/antoniocarlosrodriguezbajo/diceR_plus

[46] F. Abedinzadeh Torghabeh, Y. Modaresnia, and S. A. Hosseini, "Auto-UFSTool: An Automatic Unsupervised Feature Selection Toolbox for MATLAB," *Journal of AI and Data Mining*, 2023.

[47] H. Bengtsson, "R.matlab: Read and Write MAT Files and Call MATLAB from Within R," R package version 3.7.0, Jan. 2025. [Online]. Available: https://github.com/HenrikBengtsson/R.matlab