Clustering Ensemble: An Empirical Study

Ms.Urvashi Soni[1], Dr. Sunita Dwivedi[2]
1 Assistant professor, SAGE University Indore, 2 Associate Professor, MCRPV Bhopal

ABSTRACT

Clustering is a type of unsupervised learning that seeks to uncover the natural groupings of a set of patterns, points, or objects. A fundamental issue in clustering algorithms is the lack of a deterministic technique based on which users may pick which clustering method best matches a particular set of input data. This is because particular optimization criteria are used. Clustering ensemble as knowledge reuse provides a solution to the clustering difficulties. It tries to investigate high stability and robustness findings by assembling computed solutions obtained by base clustering algorithms without access to the features. When low-quality ensemble members are used, combining base clustering's reduces the quality of the final solution. Several academics in this field have proposed the idea of clustering ensemble selection to pick a subset of base clustering based on quality and variety. While the clustering ensemble combines all ensemble members, the clustering ensemble selection selects a subset of those members to build a smaller cluster ensemble that outperforms the clustering ensemble. This examination covers the history of data clustering, as well as an overview of basic clustering techniques, clustering ensemble algorithms, including ensemble generation mechanisms and consensus functions, and clustering ensemble selection strategies that take quality and variety into account.

1. INTRODUCTION

Clustering, in general, analyses and categorises available data based on particular criteria, with the same samples in one category finding the greatest disparities with samples in other categories. Pattern recognition, statistics, bioinformatics, data mining, and machine learning (Eisen et al., 1998; Shi and Malik, 2000; Arbelaez et al., 2011; Sîrbu et al., 2012; Zarikas et al., 2020) are just a few of the fields where clustering has been used. The connection and subsequent grouping are calculated without any prior knowledge of the class label patterns. Clustering algorithms maximise the global objective function based on specific criteria such pattern similarity (Jain, 2010; Sharma and Seal, 2020). The clustering problem is partitioned appropriately by partitioning the data points into clusters, and the results are assessed using the objective function to produce the final clusters. Data clustering is categorised as an essential, ill posed problem because it is based on unsupervised learning (Barthélemy and Brucker, 2001; Drineas et al., 2004). As a result, maximising the objective function in order to attain global results is impossible. Although there are numerous approaches to discovering appropriate solution that are often adequate but not always the best (including heuristic methods), most clustering methods can just resolve to a local optimum. Whenever datasets possess spatial forms, they can converge to a global optimum in some instances. For example, in circumstances when the clusters are adequately isolated with the same sizes, k-means has the ability to converge to the global optimum (Jain, 2010). It's tough to find a clustering algorithm that works for all datasets. As a result, various clustering techniques (Kleinberg, 2002; Han et al., 2011; Sinaga and Yang, 2020; Li et al., 2020) have been presented. The researchers introduced the idea of consensus clustering (cluster ensembles) to overcome this problem (Strehl and Ghosh, 2003). Consensus clustering, also known as clustering ensemble (CE), combines many clustering results into a resultant cluster without obtaining access to the techniques or features. (Fred and Jain, 2002; Faceli et al., 2007; Yu et al., 2013; Rafiee et al., 2013; Li et al., 2017; Wu et al., 2018; Hamidi et al., 2019). To reach the final solution, the consensus function is employed to combine the clustering. Traditionally, to arrive at the final result, all of the basic clustering (clustering members) are joined using a consensus function. Keep in mind that clustering obtained in this manner may or may not provide genuine benefits to the clustering ensembles' eventual solution.(Hong et al., 2009; Azimi and Fern, 2009a).

Researchers have recently proposed that a subset of clustering solutions be chosen to improve clustering outcomes (Fern and Lin, 2008; Lu et al., 2013; Alizadeh et al., 2014b; Yousefnezhad et al., 2016; Shi et al., 2018; Abbasi et al., 2019). Clustering ensemble selection is the name for this method. The goal of the selection approach is to choose the best clustering members from among the base clustering. The basic goal of cluster ensemble selection (CES) is to pick an acceptable subset of base clustering (BC) and combine them into a smaller cluster ensemble that outperforms the set of all base clustering (Fern and Lin, 2008; Azimi and Fern, 2009a). Clustering ensembles have numerous uses in a variety of industries, including engineering. Transportation (Xiao et al., 2016), image processing (Liu et al., 2018; Wu et al., 2017), remote sensing (Yao et al., 2017; Sarkar et al., 2019), malware detection (Chakraborty et al., 2017; Zhang et al., 2017), time series analysis (Ramasso et al., 2015; Stolz et al (Zhao et al., 2016). Xiao et al. (2016), for example, introduced a clustering ensemble method and used it to diagnose faults in high-speed train running gear.

Chakraborty et al. (2017) and Zhang et al. (2017) developed an intelligent malware categorization system that employs the clustering ensemble method to detect and combat malware. To quantify damage in composites, Ramasso et al. (2015) used a clustering ensemble technique in acoustic emission time-series. Stolz et al. (2020).presented a monitoring network to locate similar and redundant air quality stations using the clustering ensemble method. Zhao et al. developed a clustering ensemble strategy to tackling the problem of resource allocation in business process management (2016).

2. TECHNIQUES FOR BASIC CLUSTERING

From a broad perspective, clustering methods can be divided into two categories: partition approaches and hierarchical methods. As final clusters, partition algorithms return a single clustering. While hierarchical approaches, which include agglomerative and divisive algorithms, produce nested clusters of datasets, they return a gradient of clustering. Each point (pattern) is treated as a cluster using agglomerative algorithms, and the pair of nearest clusters in the current clustering is selected and combined into one to form the next clustering. Divisive algorithms, unlike agglomerative algorithms, construct a chain of clusters at every step and then break the suitably chosen cluster into two smaller clusters.

Basic clustering techniques are some of the clustering algorithms that are employed in many advanced clustering approaches like clustering ensemble. The k-means algorithm is a well-known partition technique that is reasonably basic and hence utilised more frequently by academics. Single-link (Sibson, 1973), Complete-link (King, 1967), and Average-link (King, 1967) are the most widely used basic algorithms in hierarchical algorithms (Olson, 1995).

X = x1, x2... x_n, where xi = (xi1... xid), 2 d, I = 1... n, given a set of n objects in d-dimensional space. The clustering algorithm determines a partition of the data into k clusters that meets a specified goal, which is stated in terms of a similarity (distance) measure d (xi, xj). A clustering of X is k groupings of objects as = c1, c2... ck, where ci is just a cluster of clustering, I = 1, 2... k; and k i=1 ci = X. k i=1 ci X in the context of an outlier.

Partitional clustering

The objective of the partitioning clustering algorithms is to divide the dataset into numerous clusters so that points inside one cluster are more similar to each other than those in other clusters. A dataset is separated into k groups in the partitioning clustering process. The hard partitioning must satisfy two requirements: (1) each cluster must include at least one point, and (2) each point must belong to only one cluster. The search for all possible partitions over a dataset, which is required for an optimization techniques, can be prohibitively time-consuming. K-means is the most often used local optimization approach (MacQueen et al., 1967). The k-means clustering method is a partitional clustering method that uses solely the numerical dataset (MacQueen et al., 1967). The k-means algorithm is based on the premise that k clusters are developed in such a way that the centroids of their own clusters are closer than the centroids of other clusters. The algorithm's operation begins with the selection of k points as the centroid. Following the selection of the k points, all points will be assigned to the nearest centroid, resulting in the formation of k clusters. After that, a centroid will be calculated by taking the average of each cluster's points. The mean vector is made up of these centroids, with each field of the vector equalling a cluster centroid. The mean vector is made up of these centroids, with each field of the vector equalling a cluster centroid. Finally, the new centroid creates additional clusters as a result of this process. The process will be completed if the cluster centres do not change. Figure 1 shows an example of k means clustering with various k values. Algorithm 1 (Jain et al., 1988) shows the steps of the k-means algorithm:

1st Algorithm k-means method was used to investigate k clusters.

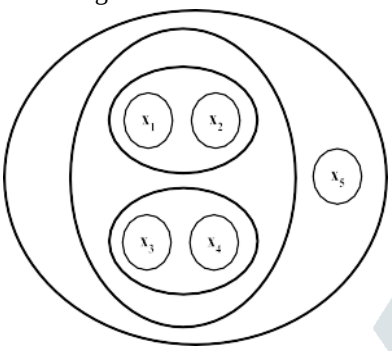
1. Assign k points as the initial centroids.
2. Assign all points to the centroid that is closest to them.
3. Recalculate each cluster's centroid.
4. Check to see whether the centroids haven't moved by repeating the procedures 2 and 3.

Fig. 1. An example of k-means clustering

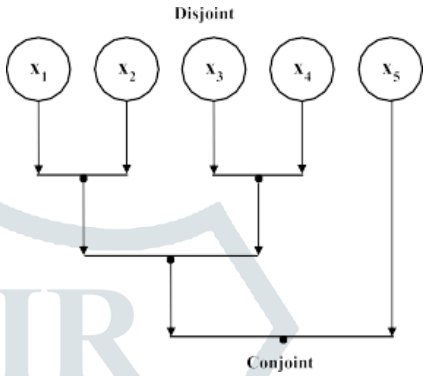
Since the vectors are only stored, storage space must be O (n d), where n is the number of points and d is the number of characteristics (dimension). Aside from that, the time indicators appear as O (I k n d), where I denotes the number of iterations required for the positioning process.

Clustering in hierarchical order

In a hierarchical clustering technique, a dendrogram can be used to show the processes that occur. This tree-like figure depicts hierarchical clustering in an accessible manner. It is made up of several layers of nodes, each of which represents a clustering.



Clustering



Dendrogram

Fig. 2

A clustering is obtained by severing the dendrogram at the relevant layer. Figure 2 depicts a clustering example and its dendrogram.

Agglomerative hierarchical clustering is when a hierarchical clustering algorithm builds a dendrogram from the bottom up, while divisive hierarchical clustering is when the method builds a dendrogram from the top down (Guénoche et al., 1991). The single-link clustering (S-link), complete-link clustering (C-link), and average-link clustering (A-link) algorithms are all well-known hierarchical clustering approaches

In natural data, different single clustering techniques find diverse clusters. This is because each clustering method includes a set of optimization requirements. Furthermore, because the data are unlabelled, a same method with different parameters (e.g., the k-means algorithm with varying values of k) or initialization data order finds different clusters. Finding the true number of clusters automatically is a difficult task in data clustering. While the data is unlabelled, internal criteria are used to determine the true number of clusters. Clustering techniques separate the dataset into clusters without accessing labels when class labels are available, and then external criteria are used to evaluate the clustering result.

3. Criteria for evaluation

Due to the lack of a supervisor, the clustering evaluation is crucial. The bulk of cluster validity metrics in the literature can be divided into two categories: internal and external criteria.

3.1. Internal quality measures

Internal criteria characterize the data's basic structure imposed by the clustering algorithm by determining the best match (based on a predetermined measure) here between data and the underlying structure (clustering) without using a reference partition.

Measures like:
Dunn Index (DI) Davies–
Bouldin Index (DBI)PBM
Index
Silhouette Index (SI)

3.2. External Quality Measures

External criteria can be used to determine the difference in structure between a clustered structure and a structure defined by class labels. The clustering results are evaluated using external criteria.

Measures like:
Normalized Mutual Information (NMI)
Adjusted Rand Index (ARI)
Disagreement and Agreement Index (DAI)
F-measure (FM)

These performance metrics, particularly NMI, are well-known and widely used in the literature. (Fern and Brodley, 2004; Fred and Jain, 2005; Fern and Lin, 2008; Ayad and Kamel, 2008, 2010; Jia et al., 2011; Franek and Jiang, 2014; Yu et al., 2014; Alizadeh et al., 2014a; Abbasi et al., 2019)

Internal criteria are used to determine the true number of clusters when the data is unlabelled. In most cases, a single clustering algorithm is run with different numbers of clusters, and then the appropriate number of clusters is chosen based on internal criteria. For real-world datasets like microarray datasets, many researchers discovered the number of actual clusters, k

4. Consensus clustering

A tried-and-true method in supervised learning is the ensemble of classifiers. Consensus clustering and clustering ensembles are the two end goals of unsupervised learning. The goal is to combine different viewpoints on the data to create a more stable partition. Given a collection of divisions of identical data x: with p = "p1, p2... P n," p1 = "c11, c12... C1k1," and pn = "c1, cn2... Cn kn," you can create a new partition that utilizes the data from all n partitions.

Consensus clustering goals:

Robustness: In some ways the combination performs better than each separate segment

Consistency: The combination resembles each partition alone.
Stability: Because of its reduced sensitivity to noisy and outliers, in the resulting partition
Novelty: The combination can produce alternative partitions that the clustering techniques that produced the separate partitions are unable to produce.
Consensus clustering advantages:
Prior partitions utilizing the same or different attributes can be employed because the consensus can be computed from the partition assignments. Individual partitions can be obtained independently using distributed computing. Only the individual partition assignments are required for consensus in terms of privacy.

4.1. Process of consensus

Consensus clustering typically consists of two steps:
Create the separate divisions that will be joined.
Join the partitions, in order to get final partition.

4.2 Partition generation

Different example representations: Diversity by generating partitions with different subsets of attributes.
Different clustering algorithms: Take advantage that all clustering algorithms have a different biases
Parameter initialization: use clustering algorithms able to produce different partitions using different parameters
Subspace projection: use dimensionality reductions techniques
Subsets of examples: use random subsamples of the dataset (bootstrapping)

4.3 Consensus generation

Co-occurrence based methods: use the labels obtained from each individual clustering, and the coincidence of the labels for the examples
Relabeling and voting, co-association matrix, graph and hyper graph partitioning, information theory measures, finite mixture models.
Median partition based methods: given a set of partitions (p) and a similarity function (γ (pi, pj)), find the partition (pc) that maximizes the similarity to the set:

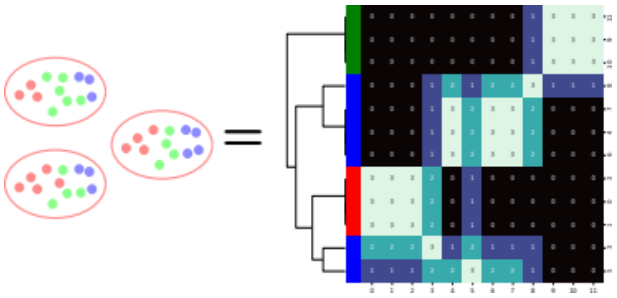
$$pc = \arg\max_{p \in px} \sum_{pi \in p} \gamma(p, pi)$$

4.4 Co-occurrence based methods

Voting and labelling
Fix the labelling correspondence issue first.
Then, using various voting techniques, ascertain the consensus.
Hornik and dimitriadou weingessel voting-merging: a clustering ensemble method
1. Create a clustering
2. Verify whether the current consensus corresponds.
3. Based on their cluster assignment, each example receives a vote.
4. Modify the general consensus

4.5. Co-association matrix

Evaluate a number with which two samples appear in the same cluster using the co-association matrix. Use the matrix to identify new or similar qualities. Information from the co-association matrix should be subjected to a cluster method.



Fred finding consistent clusters in data partitions multiple classifier systems, 2018
Fred, jain combining multiple clustering using evidence accumulation IEEE trans. Pattern anal. Mach. Intell., 2019,
1. Compute the co-association matrix
2. Apply hierarchical clustering (different criteria)
3. Use a heuristic to cut the resulting dendrogram

Graph and hyper graph partitioning
Define consensus as a graph partitioning problem. Different methods to build a graph or hyper graph from the partitions Strehl, Ghosh Cluster ensembles- A knowledge reuse framework for combining multiple partitions Journal of Machine Learning Research, MIT Press, 2018.

- Cluster based Similarity Partitioning Algorithm (CSPA).
- Hyper Graph-Partitioning Algorithm(HGPA)
- Meta-Clustering Algorithm (MCLA)

CSPA: Compute a similarity matrix from the clustering's.
Hyper edges matrix: For all clustering, compute an indicator matrix (H) that represents the links among examples and clusters (Hyper graph) Compute the similarity matrix as: S = 1 r HHT where r is the number of clustering. Apply a graph partitioning algorithm to the distance matrix (METIS). Drawback: Quadratic cost in the number of examples O (n 2kr)
CSPA: example

	C_1	C_2	C_3
x_1	1	2	1
x_2	1	2	1
x_3	1	1	2
x_4	2	1	2
x_5	2	3	2

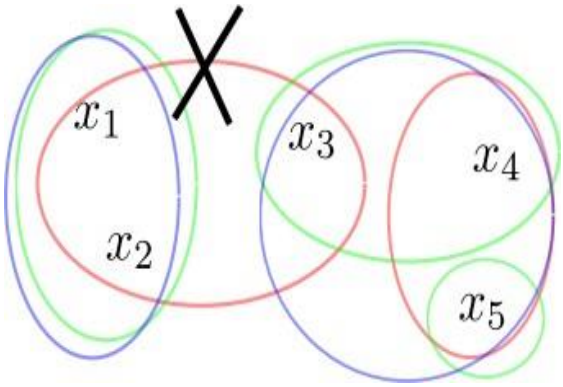
⟹

	$C_{1,1}$	$C_{1,2}$	$C_{2,1}$	$C_{2,2}$	$C_{2,3}$	$C_{3,1}$	$C_{3,2}$
x_1	1	0	0	1	0	1	0
x_2	1	0	0	1	0	1	0
x_3	1	0	1	0	0	0	1
x_4	0	1	1	0	0	0	1
x_5	0	1	0	0	1	0	1

$S =$

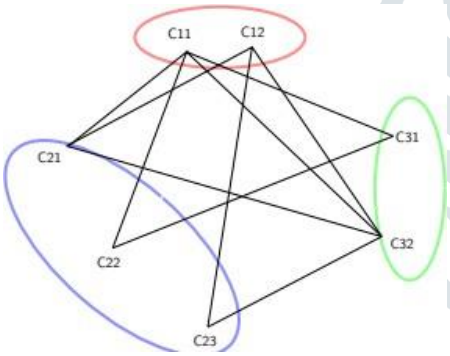
	x_1	x_2	x_3	x_4	x_5
x_1	1	1	1/3	0	0
x_2	1	1	1/3	0	0
x_3	1/3	1/3	1	2/3	1/3
x_4	0	0	2/3	1	2/3
x_5	0	0	1/3	2/3	1

HGPA: Partitions the hyper graph generated by the examples and their clustering. The indicator matrix is partitioned into k clusters of approximately the same size. The HMETIS hyper graph partitioning algorithm is used. Linear in the number of examples O (nkr)



MCLA: Group and collapse hyper edges and assign the objects to the hyper edge in which they participate the most.
Algorithm
1. Build a meta-graph with the hyper edges as vertices (edges have the vertices similarities as weights, Jaccard)
2. Partition the hyper edges into k Meta clusters 3. Collapse the hyper edges of each Meta cluster 4. Assign examples to their most associated Meta cluster. Linear in the number of examples O (nk2 r 2)

	$C_{1,1}$	$C_{1,2}$	$C_{2,1}$	$C_{2,2}$	$C_{2,3}$	$C_{3,1}$	$C_{3,2}$
x_1	1	0	0	1	0	1	0
x_2	1	0	0	1	0	1	0
x_3	1	0	1	0	0	0	1
x_4	0	1	1	0	0	0	1
x_5	0	1	0	0	1	0	1



Information Theory: Information Theory measures are used to assess the similarity among the clusters of the partitions. For instance: ◦ Normalized Mutual Information ◦ Category Utility. The labels can be transformed to a new set of features for each example (measuring example coincidence). The new features can be used to partition the examples using a clustering algorithm.

Finite Mixture Models: The problem is transformed into the estimation of the probability of assignment. The mixture is composed by the product of multinomial distributions, one for each clustering. Each example is described by the set of assignments of each clustering. An EM algorithm is used to find the probability distribution that maximized the agreement.
Median partition based methods: Median Partition Methods. Given a set of partitions (P) and a similarity function among partitions Γ(Pi , Pj), the Median Partition Pc is the one that maximizes the similarity to the set Pc = arg max P ∈Px X Pi∈P Γ(P, Pi) } Has been proven to be a NP-hard problem for some similarity functions Γ.

Similarity functions: Based on the agreements and disagreements of pairs of examples between two partitions ◦ Rand index, Jaccard coefficient, Mirkin distance (and their randomness adjusted versions). Based on set matching
◦ Purity, F-measure
Based on information theory measures (how much information two partitions share)
◦ NMI, Variation of Information, V-measure
Strategies: Best of k (the partition of the set that minimizes the distance). Optimization using local search: Hill Climbing, Simulated Annealing, Genetic Algorithms
◦ Perform a movement of examples between two clusters of the current solution to improve the partition. Non Negative Matrix Factorization ◦ Find the partition matrix closest to the averaged association matrix of a set of partitions

Table 1 Comparison of CE approaches based on consensus function

CE Approach	Advantages	Disadvantages	Description
Co-association matrix	• No need the specification of the number of clusters in the consensus partition.	• A quadratic computational complexity. • There is no guidance for chosen the clustering algorithm for final clusters ofcombination of clusters. • A combination with a small number of partitioning may be does not provide a reliable estimate of the amount of Correlation.	• Usually, hierarchical algorithms, such as S-link, A-link and C-link, are used for combining results by correlation matrix.
Graph and Hyper Graph partitioning	• Computational complexity for Cluster-based hyper graph partitioning is low. • There are different ways to representation of vertices and edges such as each vertex can be represented by object or cluster or both and each edge can be represented by similarity between two objects or two clusters. • ability to handle missing data	• These algorithms act properly for balanced clusters. • They need the specification of the number of clusters in the consensus partition.	• Three CSPA, HGPA, and MCLA are the famous methods in this approach, which arecompared with many of others methods
Relabeling and voting	• The approach is usually easy to understand and implement.	• It requires a relatively larger number of clusterings to obtain reliable result. • Correspondence problem can be solved in most cases with certain accuracy if all partitions have the same number of clusters. • They have high computational cost.	Contrary to other approaches, the approach solves the correspondence problem.
Mutual information	• Computational complexity of the algorithm is low. • Complete avoidance of solving the label correspondence problem • ability to handle missing data	• This method requires to be restarted several times to avoid the convergence tolow quality local minima.	This approach will be considered each output of clustering algorithm as a categorical feature. In this process, the L features can be considered as an intermediate feature space on which other clustering algorithm can work.
Finite mixture model	• Computational complexity of the algorithm is low. • Complete avoidance of solving the label correspondence problem • ability to handle missing data	• Adjusting the primary parameters is hard. • It can be seen that those clustering's that are less sensitive to those perturbations are more likely to happen.	·The main assumption is that the labels (label assigned to the object) are modeled as random variables drawn from a probability distribution described as a mixture of multivariate component Densities.

5. Conclusion

Overall, cluster formation algorithms such as k-means are simple to use, but they are only appropriate for certain datasets (convex shape). They, on the other hand, form clusters with a poor degree of accuracy. Hierarchical approaches, on the other hand, have a high level of complexity (O (n 2) but yield very accurate clusters. Furthermore, these approaches lack a global optimization objective function. Many partitional and hierarchical clustering techniques use input parameters like k-means to influence the outcome. If these settings are chosen incorrectly, the outcomes will be poor. Each approach focuses on a specific type of data. As a result, they have a high level of efficiency when it comes to specific data. Even using the best clustering methods with the optimal parameter values is not appropriate for all datasets. For huge datasets with an outlier, even the most powerful clustering approaches provide poor results. Developing a clustering approach that yields low-complexity results and offers reliable solutions for every type of datasets has remained a difficulty.

By aggregating the findings of numerous clustering, clustering ensemble techniques can effectively raise the accuracy and stability levels. There are two major issues with clustering ensembles: diversity and consensus function. The raw data is not available in knowledge reuse issues. When a small number of ensemble members with low diversity are used as the input to the consensus function, the result is a final solution of poor quality. As a result, developing new quality variety with a limited number of base clustering is a difficulty.

The size of the ensemble, the quality of the base clustering, and the diversity of the data all affect the consensus functions. Some consensus functions, such as HGPA, are more sensitive to variety than others, such as CSPA. It's difficult to come up with a consensus function that's less sensitive to ensemble size, quality, and diversity.

The purpose of cluster ensemble selection is to improve CE performance. However, there were certain flaws in the CES method, such as: Because BC labels are virtual, and this approach discovers a subset of a huge library of BC, the size of a suitable subset of BC is unclear, and the proposed diversity measures are informally defined. In this instance, if the set of BC is tiny, the CES cannot improve the CE result. Consensus function, on the other hand, is a significant aspect in enhancing BC. The link between diversity and base clustering quality is still unknown.

Based on the review done in this paper, the problems in consensus function include accuracy, complexity, estimating the number of true clusters, and obtaining outlier.

6. References

1. Abbasi, S.-o., Nejatian, S., Parvin, H., Rezaie, V., Bagherifard, K., 2019. Clustering ensemble selection considering quality and diversity. Artif. Intell. 1340.
2. Aggarwal, C.C., 2021. Re-designing distance functions and distance-based a p p l i c a t i o n s for high dimensional data. ACM SIGMOD Agrawal, A.K., Goldfarb, A., 2016. Restructuring Research: Communication Costs and the Democratization of University Innovation. Technical Report, National Bureau of Economic Research.
3. Akbari, E., Dahlan, H.M., Ibrahim, R., Alizadeh, H., 2015. Hierarchical cluster ensemble selection. Eng. Appl. Artif. Intell.
4. Alizadeh, H., Minaei-Bidgoli, B., Parvin, H., 2014a. Cluster ensemble selection based on a new cluster stability measure. Intell. Data Anal. Alizadeh, H., Minaei-Bidgoli, B., Parvin, H., 2020b. To improve the quality of cluster ensembles by selecting a subset of base clusters. J. Exp. Theor. Artif. Intell.
5. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J., 2021. Contour detection and hierarchical image segmentation. IEEE Trans. Pattern Anal. Mach. Intell.
6. Ayad, H.G., Kamel, M.S., 2018. Cumulative voting consensus method for partitions with variable number of clusters. IEEE Trans. Pattern Anal. Mach. Intell.
7. Ayad, H.G., Kamel, M.S., 2019. On voting-based consensus of cluster ensembles. Pattern Recognit.
8. Azimi, J., Fern, X., 2009a. Adaptive cluster ensemble selection. In: Proceedings of the 21st International Jont Conference on Artifical Intelligence, Vol. 9. Morgan Kaufmann Publishers Inc.,
9. Azimi, J., Fern, X., 2019b. Adaptive cluster ensemble selection. In: International Joint Conferences on Artificial Intelligence
10. Bae, E., Bailey, J., 2016. Coala: A novel approach for the extraction of an alternate clustering of high quality and high dissimilarity. In: Sixth International Conference on Data Mining (ICDM). IEEE,.
11. Bagherinia, A., Minaei-Bidgoli, B., Hosseinzadeh, M., Parvin, H., 2020. Reliability-based fuzzy clustering ensemble. Fuzzy Sets and Systems.
12. Bailey, T.L., Elkan, C., et al., 1994. Fitting a mixture model by expectation maximizationto discover motifs in bipolymers. In: Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology, Vol. 2,.
13. Barthélemy, J.-P., Brucker, F., 2011. NP-Hard approximation problems in overlapping clustering. J. Classification.
14. Beauchemin, M., 2015. A density-based similarity matrix construction for spectral clustering. Neurocomputing.
15. Berikov, V., 2014. Weighted ensemble of algorithms for complex data clustering. Pattern Recognit. Lett.
16. Berikov, V., Pestunov, I., 2017. Ensemble clustering based on weighted co-association matrices: Error bound and convergence properties.
17. Bobrowski, L., Bezdek, J.C., 1991. C-means clustering with the l l and l norms. IEEE Trans. Syst. Man Cybern
18. Brown, R.D., Martin, Y.C., 1996. Use of structure- activity data to compare structure-
19. Based clustering methods and descriptors for use in compound selection. J. Chem. Inf. Computer. Sci.
20. Caruana, R., Elhaway, M., Nguyen, N., Smith, C., 2006. Meta clustering. In: Sixth International Conference on Data Mining, ICDM. IEEE,
21. Chakraborty, T., Pierazzi, F., Subrahmanian, V., 2017. Ec2: Ensemble clustering and classification for predicting android malware families. IEEE Trans. Dependable Secure Computer.
22. Correa-Morris, J., 2013. An indication of unification for different clustering approaches.
23. Davies, D.L., Bouldin, D.W., 2016. A cluster separation measure. IEEE Trans. Pattern Anal. Mach. Intell.

24. Deb, K., Agrawal, S., Pratap, A., Meyarivan, T., 2000. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In: International Conference on Parallel Problem Solving from Nature. Springer,
25. Domeniconi, C., Gunopulos, D., Ma, S., Yan, B., Al-Razgan, M., Papadopoulos, D., 2017. Locally adaptive metrics for clustering high dimensional data. Data Min. Knowl. Discov.
26. Drineas, P., Frieze, A., Kannan, R., Vempala, S., Vinay, V., 2004. Clustering large graphs via the singular value decomposition. Mach. Learn.
27. Dudoit, S., Fridlyand, J., 2003. Bagging to improve the accuracy of a clustering procedure. Bioinformatics
28. Dunn, J.C.,. A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. J. Cybern
29. Dunn, J.C., 2014. Well-separated clusters and optimal fuzzy partitions. J.Cybern.
30. Eisen, M.B., Spellman, P.T., Brown, P.O., Botstein, D., 1998. Cluster analysis and display of genome-wide expression patterns. Proc. Natl. Acad. Sci...
31. Elghazel, H., Aussem, A., 2013. Unsupervised feature selection with ensemble learning.
32. MachFaceli, K., De Carvalho, A.C., De Souto, M.C., 2007. Multi-objective clustering ensemble. Int. J. Hybrid Intell. Syst.
33. Fern, X.Z., Brodley, C.E., 2013. Random projection for high dimensional data clustering: A cluster ensemble approach. In: Proceedings of the Twentieth International Conference on International Conference on Machine Learning,