INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

Cluster Predictive Model Using Affinity Propagation Algorithm to Group Mushroom 5.8s rRNA Sequences

P Sudhasini[1,2]*, B Ashadevi[3]

1 Research Scholar, Department of Computer Science, Mother Teresa Women's University, Kodaikanal, Tamilnadu, India
2 Assistant Professor, Lady Doak College, Madurai, India
3 Assistant Professor, Department of Computer Science, M.V. Muthaiah Govt. Arts College for Women, Dindigul, Tamilnadu, India

OPEN ACCESS

Received: 31-07-2022
Accepted: 02-10-2022
Published: 03-11-2022

Citation: Sudhasini P, Ashadevi B (2022) Cluster Predictive Model Using Affinity Propagation Algorithm to Group Mushroom 5.8s rRNA Sequences. Indian Journal of Science and Technology 15(41): 2129-2142. https://doi.org/10.17485/IJST/v15i41.1341

* **Corresponding author**.

sudhabaskee@gmail.com

Funding: None

Competing Interests: None

Copyright: © 2022 Sudhasini & Ashadevi. This is an open access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Background: The main emphasis of the article is biological information on a distinct species of mushroom (Phylum Basidiomycota) data collection of 5.8s rRNA sequences. Macrofungi from the phylum Basidiomycota are predominantly used as therapeutic mushrooms in several countries. During the rainy season, hundreds of macrofungal basidiocarps were discovered in Tamilnadu. The internal transcribed spacer (ITS) and 5.8S rRNA gene sequence markers, which have been collected from NCBI, were used to isolate at least thirty of these strains that fall under the Basidiomycota kingdom (suborders of Polyporales, Hymenochataeles, and Russuales), which have the therapeutic properties of the Basidiomycota kingdom. **Objectives:** This article's main objective is to organise the sequences according to similarity utilising multiple sequence alignment and an algorithmic perspective. **Methods:** In this paper, we use 30 × 30 pairwise similarity matrix data of these thirty 5.8s rRNA mushroom sequences obtained using the clustal omega tool to develop an affinity propagation approach. As a continuation of earlier work, this will be evaluated against k-means, hierarchical clustering based on the ideal cluster, and time and space complexity. **Findings:** The affinity propagation algorithm typically discourages providing the initial number of clusters; therefore, the optimal number of cluster values and grouping of clustered results obtained from the affinity propagation algorithm are also the same as the results obtained from the previous existing research work using the k-means, hierarchical agglomerative clustering algorithm. **Novelty:** The overall suggested technique involves applying the cluster validation metrics Silhouette score, Calinski-Harabasz Index, and Davies-Bouldin Index methodologies to find the ideal number of clusters. The CD-hit Clustering tool does not offer these metrics, and the Cluster Omega tool does not support this kind of extension work. This follow-up work assists bioinformatics researchers in obtaining favourable results by utilising the existing software prior to working in wet laboratories; rather than wasting a lot of chemical resources, this result

will open the door for a targeted approach.

Keywords: Affinity propagation; Cluster metrics; Kmeans; mushroom sequences; Bioinformatics; Data science; Silhouette score; Calinski-Harabasz index; Davies-Bouldin index

1 Introduction

We are evolving huge datasets every day in all research fields, particularly in the field of bioinformatics. The way the data increases at the same time, as a researcher, we should also have the responsibility to take care of enormous data in a meaningful way. It should be dealt with proper care to get the appropriate insights. The data can be in any format, either structural or non-structural. We should adapt the techniques behind them to the nature of datasets. In this scenario, we are moving towards data science that is well suited for any discipline of research. Techniques like classification, regression, and clustering are inherited in data science [(1)]. The classification technique that is used to classify the dataset according to the label assigned by default. The regression technique gives a lead statistical point of view for an independent and dependent single or multivariable data point relationship. The clustering technique focuses on grouping of information without having any predetermined labels or categories to split the dataset into. This grouping deals with the unsupervised learning technique by only analysing the common characteristics of a given dataset. Nowadays, in the field of bioinformatics, we are hopefully able to perform bioinformatics research like genome analysis [(2)] using various available tools like clustal omega, Blast, and CD-hit for sequence multiple alignment and clustering [(3)]. The clustal omega tool is one of the great resources to perform multiple sequence alignment in a fast and accurate way. It displays the result as a guided tree based on a pairwise distance matrix and clustering using the bi-section k-means algorithm, but it is currently processed via command prompt and there is no GUI-based online tool available [(4)]. By utilising BLAST, clustal omega, and SWISS-model for silico-based bioinformatics projects instead of wet lab studies and by avoiding the use of unneeded chemical components, it is possible to operate efficiently throughout the lockdown time and highlight the significance of bioinformatics tools [(5)]. In this research article, we have focused on clustering thirty 5.8s rRNA mushroom sequences (under the division Basidiomycota) around Tamilnadu retrieved through NCBI. The grouping of mushroom sequences has been carried out by the affinity propagation algorithm by having the PIM (pairwise identify matrix) result obtained from the clustal omega tool. The same samples have already been implemented and published using the k-means algorithm. We have extended the previous work by implementing an affinity propagation algorithm to verify the result against the previous work. Also, some sets of tools are available to group sequences either in protein or nucleotide form, especially cd-hit, which is the most widely available tool for bioinformatics researchers. When compared with this CD-hit tool [(6)], it does not have the option to produce an optimal number of clusters, but our present research will give good comparative and standard results over k-means, hierarchical, and affinity propagation clustering, especially the optimal number of clusters for a given input dataset.

2 Literature Review

The experts talked on a variety of large data management issues, especially in the discipline of bioinformatics where data handling is crucial. For precise protein and nucleotide sequence alignment, the researchers largely employed online, publicly accessible bioinformatics tools including BLAST, FASTA, Clustaw/omega, Muscle, and Mafft [(7)]. The study effort was done to identify a resemblance between meta data strings, with the information being obtained from NCBI and EBI, from the standpoint of

biological thought on taking care of not just the sequences. The Levenshtein Edit distance technique for string-based distance computation, specifically for soft matching of words[8], was used to get the meta data attribute names in order to obtain the distance similarity value between the attributes. When dealing with large data sets, the data may not be practicable or may not be of sufficient quality for our research to be performed and expected to be accurate. In order to get around this problem, the article employs a synthetic minority oversampling technique (SMOTE) to generate fake data, but this suffers from the distribution of samples. By using the affinity propagation algorithm, clustering is done to fine-tune the imbalanced data in order to get around this[9]. Since it doesn't need an initial cluster value, the affinity algorithm performs more quickly than other algorithms like k-means, hierarchical clustering, etc. The affinity algorithm is the most effective way to generate the ideal number of clusters without introducing bias into the study[10]. The affinity propagation algorithm on each factor paves the way to a new approach towards various research domains. The author gave a new approach to detect social groups among school class members using the content attributes (dormitory, number, age, grade, etc.) And linking (relationship between seats) information to cluster them according to various clustering aspects like hierarchical agglomerative, spectral, and affinity propagation algorithms. In this experiment, the affinity propagation came out well to analyse the behaviour of students for quality education[11]. Similarly, when compared to similarity measures such as CNM, Info map, Louvain, and LCCD, the adaptive similarity affinity propagation algorithm implemented in social network communities to detect network pathway similarities has yielded promising results[12]. In the view of next generation sequencing, considering RNA regulatory to find the similarity between species by using rRNA, miRNA prediction analysis on identifying the potential target in human and mushroom rather than taking entire sequences is also permissible and given good similarity of measures in dealing with biological processes related to cancer, infection, and neurodegenerative diseases[13]. According to the article, clustering algorithms can incorporate the nature of unsupervised learning concepts without knowing the labels of the dataset. Therefore, on this type of dataset, they performed Silhouette score and Calinski-Harabasz Index validation metrics with the improved method of indexing values using PWI (peak weight index), which gave the anticipated result on determining the ideal number of clusters[14]. The study work recommended utilising Silhouette score and Calinski-Harabasz Index to identify the ideal number of clusters for coarse-grained representation models, and it ultimately selected the Calinski-Harabasz Index for the most optimal number of clusters[15]. The author of this research developed a hybrid intelligent system for diagnosing primary headaches. In their path of research they incorporated various methods like Calinski-Harabasz, Analytical hierarchy process, and weighted fuzy-Cmeans algorithms to give qualitative and expected experiment outcomes that help to make quality decisions for further treatments[16]. The research work carried out to identify the optimal number of clusters has used 21 different datasets which have selective features to demonstrate the similarity of each other. Then they have implemented k-means, consensus clustering, and three weighted consensus clustering methods to derive the clustering model and that has been validated using Silhouette , Calinski-Harabasz, Davies-Bouldin Index by finalizing that the weighted consensus clustering model was best to group the given dataset[17]. When working with PIMA datasets, the study effort insisted on internal validity indices to get the ideal number of clusters. Before applying Silhouette, Calinski-Harabasz, and Davies-Bouldin Index to validate the quality measure of clustering and produce a quality cluster, it was appropriately preprocessed by eliminating outliers and successfully making the feature selection[18]. In order to increase the dataset efficiency for training, the study illustrated the work nature of k-means and weighted calculation of the elken k-means method. Finally, they used the Calinski-Harabasz index to analyse the power grid business dataset and establish the appropriate number of clusters[19]. They have assembled actual datasets from the electric power system of the eastern area of Paraguay in order to group the data according to the demand as aspects of weekly, monthly, seasonal, and daily consumption. In order to create the clustered data, they used k-means and hierarchical agglomerative clustering with all five linkages, along with validation using the Silhouette, Calinski-Harabasz, and Davies-Bouldin indices. However, this method is not suitable for all features and algorithms, so they planned to incorporate additional new sets of new algorithms for improvisation[20]. The study highlighted the usefulness of employing the Davies-Bouldin index to assess cluster centres when applying the x-means algorithm to particular sets of iris datasets. They did this by running iterations for different cluster numbers to determine the k value for the x-means method[21]. By gathering the data from the Baden pusat statistic (BPS) official website, the paper clarified the educational resources offered by the village high school. They used the k-means algorithm with a variety of distance measures, including mixed measures, Bregmann divergences-mahalonobis distances, and Bregmann-Divergences-Squared Euclidean distance, as well as the validation method of the Davies-Bouldin index, where the optimise results were displayed on mixed measure distance[22]. The student thesis has been categorised using the clustering approach using the titles of 50 documents. Here, the k-means and k-medoids methods have been put into practise for comparison in terms of time complexity and for verifying the cluster using the Davies-Bouldin index[23]. The research was conducted by incorporating the k-means algorithm to group the customers according to the nature of the buying interest feature mentioned, and they have analysed the ideal number of groups to be clustered using the Davies-Bouldin index[24]. The same type of work has also been done over the analysis of car sales[25]. By employing hybrid k-means and decision tree methods, respectively, and evaluating the findings

over the Davies-Bouldin index closest smallest value among other cluster samples, the study effort determines the distribution of COVID-19 in the province of East Java[26]. When working with the iris dataset, they have demonstrated the cluster quality analysis utilising silhouette score, where the greatest value is regarded as a quality indicator of k-value[27]. Additionally, a(i), b(i) values can be replaced with in-cluster sums of squared error values while calculating the condensed silhouette score, which has better computational performance than the classic technique[28]. We will employ Silhouette, Calinski-Harabasz, Davies-Bouldin index validation metrics and affinity propagation with the comparison of k-means, hierarchical clustering algorithms to cluster the mushroom sequences in accordance with the evaluated studies in our suggested technique. When working with these kinds of biological data, which are restricted to certain common clustering tools like cd-hit, this may be one of the extension efforts.

3 Data Collection

Most nations employ macrofungi from the phylum Basidiomycota as medicinal mushrooms. In Tamilnadu, hundreds of macrofungi basidiocarps were found during the rainy season. Around that thirty of these strains were discovered using the internal transcribed spacer (ITS) and 5.8S rRNA gene sequence markers[29] that have been gathered from NCBI (National Center for Biotechnology Information. The sequences that were used for the research are listed below.

Table 1. Mushroom 5.8s rRNA (under the division Basidiomycota) sequences Data Set collected from NCBI database[29]

Identity of Mushroom 5.8s rRNA Sequences	Name of the Mushroom Sequence
KY491659.1	*Fulvifomes fastuosus* strain LDCMY43
KX957802.1	*Phellinus sp.* strain LDCMY28
KY491658.1	*Phellinus sp.* strain LDCMY23
KX957801.1	*Phellinus badius* strain LDCMY27
KY471289.1	*Ganoderma sp.* strain LDCMY12
KX957800.1	*Ganoderma sp.* strain LDCMY05
KY471288.1	*Phellinus sp.* strain LDCMY45
KX957799.1	*Ganoderma resinaceum* strain LDCMY01
KY471287.1	*Inonotus rickii* strain LDCMY52
KX957798.1	*Fulvifomes fastuosus* strain LDCMY39
KY471286.1	*Phellinus sp.* strain LDCMY 24
KY009873.1	*Ganoderma wiiroense* strain LDCMY19
KY111254.1	*Coriolopsis caperata* strain LDCMY42
KY009872.1	*Ganoderma sp.* strain LDCMY14
KY111253.1	*Ganoderma wiiroense* strain LDCMY11
KY009871.1	*Ganoderma sp.* strain LDCMY22
KY111252.1	*Fomitopsis ostreiformis* strain LDCMY21
KY009870.1	*Ganoderma sp.* strain LDCMY18
KY111251.1	*Ganoderma sp.* strain LDCMY16
KY009869.1	*Ganoderma wiiroense* strain LDCMY17
KY111250.1	*Ganoderma sp.* strain LDCMY41
KY009868.1	*Trametes elegans* strain LDCMY37
KY111249.1	*Phellinus badius* strain LDCMY36
KY009867.1	*Ganoderma wiiroense* strain LDCMY08
KX957805.1	*Phellinus sp.* strain LDCMY34
KY009866.1	*Ganoderma sp.* strain LDCMY04
KX957804.1	*Phellinus badius* strain LDCMY31
KY009865.1	*Ganoderma sp.* strain LDCMY06
KX957803.1	*Phellinus sp.* strain LDCMY29
KY009864.1	*Ganoderma wiiroense* strain LDCMY02

4 Methodology

4,1 Affinity propagation algorithm

The Affinity propagation algorithm has the perspective of clustering the data points which are in matrix format without giving any initial number of cluster values to group the given dataset. It communicates with each data point via message passing until it is satisfied with the proper and standard response from each data point. The Affinity algorithm has been fine-tuned with two important roles: exemplar and damping factor. For each data point, the exemplar means attempting to associate and make available one of its targets. The damping point specifies the constant value to make sure the interchangeable of data points is associated with this damping factor on every iteration without having any oscillation. The time at which the sender relates to one of the targets can be used as the exemplar for that datapoint. We can determine the cluster based on the same exemplar of various data points.

The following are the Two important role is carried out in message passing:

- Responsibility - r(i,k)

The responsibility matrix (r) consists of all values i in the matrix and k as an exemplar for $x_{i.,}$ To make sure how far x_k is suited to serve as an exemplar for x_i.

- Availability – a(i,k)

The availability matrix (a) consists of all values in the matrix and k as an exemplar, but it still measures how well the value of x_i is appropriate for x_k to act as an exemplar.

4.1.1 ***Steps to do for Affinity propagation working process***

- Consider the given n matrix dataset.
- Form a similarity matrix by calculating every cell by negating the sum of squares of the differences between other values of the matrix for all (i,j). The algorithm converges by choosing the minimum negative value of the entire matrix to fill all the diagonal positions of the matrix to get the smallest number of clusters.
- Next, construct the responsibility matrix by taking the highest value in the row and minus it for all the values in the row that apply to all the rows, respectively.

$$r(i,k) \leftarrow s(i,k) - \max_{k' \text{ such that } k' \neq k} \{a(i,k') + s(i,k')\} \tag{1}$$

The formula (1) is used to calculate every cell in the responsibility matrix. Here, i refers to the row and k refers to the column.

- Then form an availability matrix by framing two formulas to fill the values in the diagonal position and off-diagonal position. The diagonal position is calculated by the sum of all the values above zero except when self-responsibility. The off-diagonal position is calculated by including the self-responsibility and summing all the positive values of that column but excluding the joined responsibility of each other target values of that row and column.

$$a(k,k) \leftarrow \sum_{i' \, such \, that \, i' \neq k} max\left(0,\ r\left(i',k\right)\right\} \tag{2}$$

The formula (2) is used to fill the diagonal elements in the availability matrix. Here, i refers to the row and k refers to the column.

$$a(i,k) \leftarrow min\ \left(0,\ r(k,k) + \sum_{i' \, such \, that \, i' \neq k} max\left(0, r\left(i',k\right)\right\} \quad \right\} \tag{3}$$

The formula (3) is used to fill the off-diagonal elements in the availability matrix. Here, i refers to the row and k refers to the column.

- The criterion matrix will be formed by summing the availability matrix and responsibility matrix according to the value location.
- The highest criterion value of each row is represented as an exemplar. The rows that share the same exemplar values will be considered in the same cluster.

4.1.2 ***Implementation of Affinity propagation algorithm using Python***

The affinity propagation algorithm has been implemented in Python using sklearn packages. The following are the parameters used in the sklearn affinity propagation method.

Sklearn.cluster.AffinityPropagation(*,damping=0.5, maxiter=200, convergence_iter=15, copy =True, Preference = None, Affinity='euclidean', Verbose=False, random_state=None)

Damping: While updating the existing matrix value on every iteration, the data will be allotted based on the damping factor [0.5 to 1.0]. The aim is to maintain the current value according to the incoming value.

Max_iteration: This parameter is used to specify the maximum number of iterations to be performed to complete the algorithm effectively. By default, it has a number of 200 iterations to be done.

Convergence_iteration: It denotes, at which the convergence should be stopped while confirming that there is no change in the number of clusters that is estimated.

Copy: It has the Boolean value to make sure whether the input data should be copied for further reference or not.

Preference: Mostly larger values of preference will be taken as an exemplar If the preference value is assigned, then by default the median of input similarities will be taken as an exemplar.

Affinity: The calculation of every cell's data is measured using the Euclidean method, which uses a negative squared Euclidean distance between points.

Verbose: It is made up of Boolean parameters that are used to provide a detailed log of the entire output process, which will be useful for troubleshooting in the event of an error.

Random_state: A random integer value confirms the starting state of the algorithm process.

4.2 Cluster Validation Metrics

Obviously, supervised or unsupervised strategies may be used to interpret data in the field of machine learning. When dealing with classification, the validation for supervised learning is somewhat predetermined and relatively simple with the given labels in the start step of classification, therefore it may be predominately used to seek the outcome as being simpler than unsupervised learning. The clustering method falls under the area of unsupervised learning, in which the data cannot be understood by applying a preexisting label to the training dataset. As a result, grouping of information should be done in accordance with the nature of the dataset's arrival and the similarities between its features at this time; however, the similarities cannot already be identified by labelling. The evaluation metrics are the important role to play while dealing with any kind of algorithm to prove the integrity of the results. The evaluation metrics of clustering will be able to judge the quality of split up in the data according to the similarity measures over datapoints in the same cluster and with next related clusters, also to obtain the number of optimal clusters to be grouped for the given dataset. Here the experiment was carried out to validate existing and proposed methodologies using the cluster validation methods of silhouette score, Calinski-Harabasz Index, and Davies-Bouldin Index algorithm.

4.2.1 Silhouette Score

The data organized within the cluster may be effectively interpreted using the Silhouette score approach. A comparison between the mean of the intra-cluster and the closest cluster for the specified datapoints can be used to validate clustering. The measurement for verifying it is provided below.

- Mean-Intra Cluster — a(i)

To calculate the mean difference between each data point and the other data points in the same cluster, the data points are collected collectively as samples. The measurement is obtained using formula (4).

$$a(i) = \frac{1}{n_k - 1} \sum_{i' \in I_k} d(M_i, M_{i'}) \tag{4}$$

We can assume M_i as each data point where we have to calculate the mean distance (d) that defines the within-cluster for all the clustered groups(n_k)

- Mean-Nearest Cluster Distance- b(i)

Each data point's average distance from the nearest cluster that isn't a member of that cluster is determined. Calculating the mean distance between each M_i and the datapoints of every other cluster $C_{k'}$ is what is meant by Formula (5).

$$\partial\,(Mi\,,\,Ck) = \frac{1}{n_{k'}} \sum_{i' \in I_{k'}} d\left(M_i\,,M_{i'}\right) \tag{5}$$

As required by formula (6), the minimal mean value of summation should be considered when calculating the mean distance between other cluster data points.

$$b\,(i) = min_{k' \neq k}\, \partial(M_i, C_{k'}) \tag{6}$$

Based on the above procedure, for each data point of M_i, we must form the quotient Formula (7) as silhouette score where the results indicate between -1 to1.

$$S\,(i) = \frac{b\,(i) - a(i)}{max(a\,(i) - b\,(i))} \tag{7}$$

The value near 1 indicates the best cluster, with the data points arranged into the perfect cluster group. The value zero indicates that due to overlapping among the data points towards cluster arrangements, there is a high possibility of another cluster data point being mismatched with the existing cluster. The value below zero denotes a wrong cluster formation of datapoints which are not related to each other.

4.2.2 Calinski-Harabasz Index

The Calinski-Harabasz (CH) Index method is used to assess the cluster model when the dataset lacks a predefined ground truth label to validate it; as a result, this method will be very helpful in unsupervised learning when the clustered dataset has been formed using similarity rather than labelled features. The separation and cohesiveness of the datapoints provide the foundation for CH index validation. By measuring the distance between each datapoint inside a cluster and the cluster centroids, cohesiveness is determined. Based on the separation between local cluster centroids and global cluster centroids, data points are separated.

$$CH = \frac{\frac{\sum_{k-1}^{k} n_k \left\|C_k - C\right\|^2}{k-1}}{\frac{\sum_{k-1}^{k} \sum_{i=1}^{n_k} \left\|d_i - c_k\right\|^2}{N-K}} \tag{8}$$

Where nk is the number of datapoints for a given dataset, C_k is the centroid of the cluster (k), C denotes the global centroid of the entire cluster for all the datapoints, and N is the total number of data points for all the clusters. The final calculation of the CH index depicts the quality of the cluster. If the value is high, the cluster is aligned in the best way as it is dense. If the value is low, the cluster did not separate well enough to find the solution.

4.2.3 Davies-Bouldin Index algorithm

The Davies-Bouldin Index (DBI) used to find the average similarity of inter and intra cluster validation among datapoints. The Formula (9), denotes the difference between M_i datapoints belongs to cluster(k) and to the same cluster(k) centroid (G^k).

$$\delta_k = \frac{1}{n_k} \sum_{i \in I_k} \left| {}_i^{\{k\}} - G^{(k)} \right| \tag{9}$$

Also, the Formula (10) denotes the distance between one cluster to other cluster centroid G^k and $G^{k'}$ for all the clusters [C_k and $c_{k'}$].

$$d\left(k, k'\right) = d(G^{\{k\}}\,,\, G^{\{k'\}}\,) \tag{10}$$

$$DB = \frac{1}{K} \sum_{k=1}^{K} max_{k' \neq k} \left(\frac{\delta_k + \delta_{k'}}{d\left(k, k'\right)}\right) \tag{11}$$

Finally, DBI measures the mean value according to Equation (11), for all the clusters on finding the average similarity between inter and intra cluster centroids distances. The lowest value of DBI can give the validation result as a better clustered group to decide the optimal number of clusters.

4.3 Implementation Process

The overall goal of this research work is to cluster the 30 mushroom sequences using the Affinity propagation algorithm. Before getting into the actual algorithm process, the sequences have been measured with each other as pair-wise identity matrixes using the Clustal Omega tool. Then, using this PIM matrix, which consists of 900 datapoints, to construct the prediction model using the Affinity propagation algorithm. The same dataset was used for the previous similar work [30]. In continuation of the previous work, the proposed methodology is introduced here.

Here are the steps followed to carry out the execution part using Python to cluster mushroom sequences.

Step 1: The 30 mushroom sequences taken from NCBI were given as input to the Clustal omega tool to get the PIM (percent identity matrix) for all the datasets in a 30*30 matrix format. Total received 900 data points to form a cluster among them. Fig. 1 shows the PIM value for all 30 mushroom sequences.

Step 2: This PIM 30*30 similarity matrix is then taken into account as the CSV format to proceed in the Affinity Propagation algorithm using Python code. The entire 30*30 matrix headers are taken as attributes to formulate the cluster process that is mentioned in Fig 2.

Step 3: In Table 2, we have mentioned the Pseudocode to cluster mushroom sequences using the Affinity propagation algorithm. The damping factor should be set to 0.5 to 1 to ensure that the numeric value oscillates when assigning the exemplar and responsibility of the matrix on each iteration.

Step 4: The result has been obtained as the output of a group of clustered datasets and cluster indexes/center points for further reference.

Step 5: The results were checked against with the previous research work of k-means algorithm and hierarchical clustering algorithm to make sure the optimal number of clusters and validity of cluster indexes using elbow method, Silhouette score, Calinski-Harabasz Index, Davies-Bouldin Index

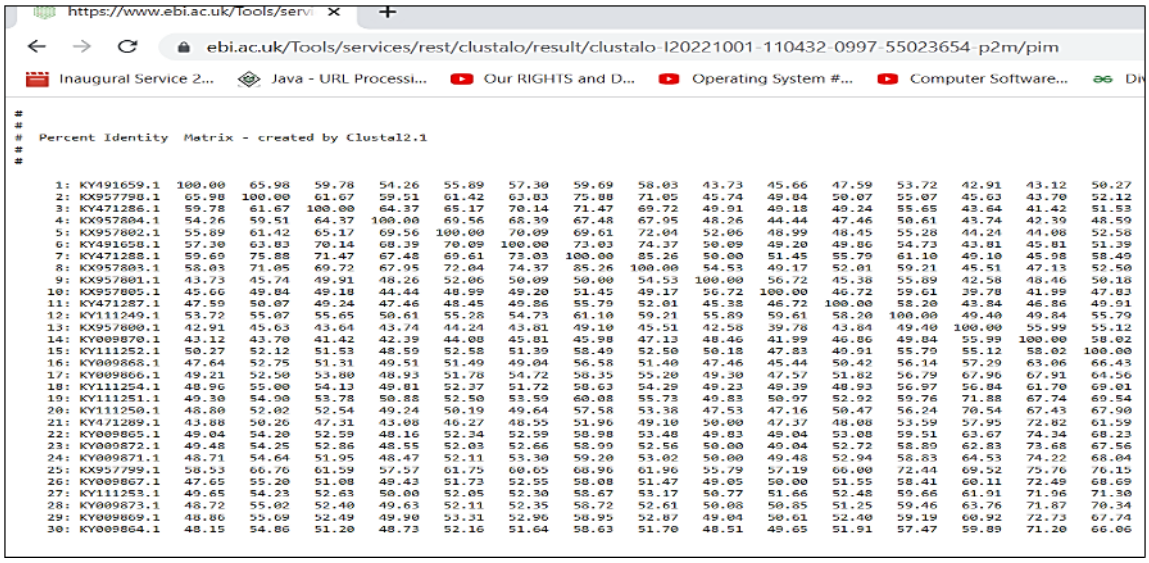


Fig 1. Percentage Identity Matrix Data set retrieved from clustal omega tool [30]

NAME	KY491659.	KX957798.	KY471286.	KX957804.	KX957802.	KY491658.	KY471288.	KX957803.	KX957801.	KX957805.	KY471287.	KY111249.
KY491659.1	100	65.98	59.78	54.26	55.89	57.3	59.69	58.03	43.73	45.66	47.59	53.72
KX957798.1	65.98	100	61.67	59.51	61.42	63.83	75.88	71.05	45.74	49.84	50.07	55.07
KY471286.1	59.78	61.67	100	64.37	65.17	70.14	71.47	69.72	49.91	49.18	49.24	55.65
KX957804.1	54.26	59.51	64.37	100	69.56	68.39	67.48	67.95	48.26	44.44	47.46	50.61
KX957802.1	55.89	61.42	65.17	69.56	100	70.09	69.61	72.04	52.06	48.99	48.45	55.28
KY491658.1	57.3	63.83	70.14	68.39	70.09	100	73.03	74.37	50.09	49.2	49.86	54.73
KY471288.1	59.69	75.88	71.47	67.48	69.61	73.03	100	85.26	50	51.45	55.79	61.1
KX957803.1	58.03	71.05	69.72	67.95	72.04	74.37	85.26	100	54.53	49.17	52.01	59.21

Fig 2. Entire 30 * 30 PIM value considered as Attributes to proceed the clustering (sample data given) [30]

Table 2. Pseudocode to cluster mushroom sequences using Affinity Propagation algorithm

#Importing all sklearn necessary packages for Affinity propagation clustering # Importing Pyplot, Principle component Analysis, numpy
1. Input the dataset (ClustalW- PIM similarity matrix values) in to the defined format as CSV file
2. Formatting dataset in to data frame by setting up as row and column
3. Define the model using Affinity propagation with damping (0.5 to 1.0)
4. Fit the model for the given dataset
5. Predict the model to assign a cluster according to the similar dataset
6. Retrieval of unique clusters
7. Retrieval of model cluster centers for reference
8. Output of number of clusters and respective dataset

5 Results and Discussion

The dataset given in PIM matrix format from the result of the clustal omega tool is passed to the Affinity propagation algorithm and executed successfully. In Tables 3 and 4, it shows the cluster indexes and cluster centers, respectively, that are fed as an array format, and in Table 4, it gives the entire result of the clustered dataset. Finally, the entire mushroom sequences are grouped into four clusters (clustered automatically without giving any initial value for clustering) according to the pairwise identity matrix. In Table 5, cluster 1 consists of 8 sequences, cluster 2 consists of 4 sequences, cluster 3 consists of 8 sequences, and cluster 4 consists of 10 sequences. A total of 30 sequences were involved in having a cluster prediction. According to various literature reviews that we had before, our research was also given good support while using this affinity propagation algorithm to make a strong assurance on clustering the mushroom sequences along with the previous research work using the Kmeans[31] and Hierarchical clustering algorithm[32] algorithms.

Table 3. Cluster indexes for all the groups

Group	Index value
Index (['KY491659.1', 'KX957798.1', 'KY471286.1', 'KX957804.1', 'KX957802.1', 'KY491658.1', 'KY471288.1', 'KX957803.1'], dtype='object')	0
Index (['KX957801.1', 'KX957805.1', 'KY471287.1', 'KY111249.1'], dtype='object')	1
Index (['KX957800.1', 'KY009870.1', 'KY111252.1', 'KY009868.1', 'KY009866.1', 'KY111254.1', 'KY111251.1', 'KY111250.1'], dtype='object')	2
Index (['KY471289.1', 'KY009865.1', 'KY009872.1', 'KY009871.1', 'KX957799.1', 'KY009867.1', 'KY111253.1', 'KY009873.1', 'KY009869.1', 'KY009864.1'], dtype='object')	3

Table 4. Cluster centers value for the four clusters according to the PIM values

[[58.03 71.05 69.72 67.95 72.04 74.37 85.26 100. 54.53 49.17 52.01 59.21 45.51 47.13 52.5 51.4 55.2 54.29 55.73 53.38 49.1 53.48 52.56 53.02 61.96 51.47 53.17 52.61 52.87 51.7]
[45.66 49.84 49.18 44.44 48.99 49.2 51.45 49.17 56.72 100. 46.72 59.61 39.78 41.99 47.83 45.44 47.57 49.39 50.97 47.16 47.37 49.04 49.04 49.48 57.19 50. 51.66 50.85 50.61 49.65]
[48.8 52.02 52.54 49.24 50.19 49.64 57.58 53.38 47.53 47.16 50.47 56.24 70.54 67.43 67.9 71.36 81.31 71.24 93.86 100. 71.57 77.57 76.63 77.45 84.7 73.75 77.31 75.12 74.5 71.62]
[48.72 55.02 52.4 49.63 52.11 52.35 58.72 52.61 50.08 50.85 51.25 59.46 63.76 71.87 70.34 72.34 76.39 77.87 80.34 75.12 83.08 89.15 89.09 91.64 93.08 95.15 97.03 100. 92.79 90.92]]

Table 5. Dataset grouped as Four Clusters

Sl. No.	Identity-Name (5.8s rRNA seq)	Cluster	Sl. No	Identity-Name (5.8s rRNA seq)	Cluster
1	KY491659.1- Fulvifomes fastuosus	**0**	**1**	KY471289.1 - Ganoderma sp	**3**
2	KX957798.1 - Fulvifomes fastuosus	**0**	**2**	KY009865.1 - Ganoderma sp	**3**
3	KY471286.1 - Phellinus sp.	**0**	**3**	KY009872.1 - Ganoderma sp	**3**

Continued on next page

Table 5 continued

4	KX957804.1 - Phellinus badius	**0**	**4**	KY009871.1 - Ganoderma sp	**3**
5	KX957802.1 - Phellinus sp	**0**	**5**	KX957799.1 - Ganoderma resinaceum	**3**
6	KY491658.1 - Phellinus sp	**0**	**6**	KY009867.1 - Ganoderma wiiroense	**3**
7	KY471288.1 - Phellinus sp	**0**	**7**	KY111253.1 - Ganoderma wiiroense	**3**
8	KX957803.1 - Phellinus sp	**0**	**8**	KY009873.1 - Ganoderma wiiroense	**3**
			9	KY009869.1 - Ganoderma wiiroense	**3**
			10	KY009864.1 - Ganoderma wiiroense	**3**
1	KX957801.1 - Phellinus badius	**1**			
2	KX957805.1 - Phellinus sp	**1**			
3	KY471287.1 - Inonotus rickii	**1**			
4	KY111249.1 - Phellinus badius	**1**			
1	KX957800.1 - Ganoderma sp	**2**			
2	KY009870.1 - Ganoderma	**2**			
3	KY111252.1 - Fomitopsis ostreiformis	**2**		Cluster Number	**Count**
4	KY009868.1 - Trametes elegans	**2**		0	**8**
5	KY009866.1 - Ganoderma sp	**2**		1	**4**
6	KY111254.1 - Coriolopsis caperata	**2**		2	**8**
7	KY111251.1 - Ganoderma sp	**2**		3	**10**
8	KY111250.1 - Ganoderma sp	**2**		Total count of sequence	**30**

The affinity algorithm results are compared with the previously published K-means algorithm [31]. In particular, the number of clusters is the same in both the algorithms, while in the implementation of k-means clustering, we obviously must pass the initial cluster value to partition the dataset at the initial stage of the algorithm. Here we have done with the elbow method by incorporating sum of squared error derivation for calculating distance among all the datapoints of every number of clusters we have given for validation. Then we had the elbow point pitched at one value and moved on slightly straight. Here the optimal number of clusters is shown in Figure. 3 the elbow method given the number of cluster elbow point prediction as 4. Similar to this, we compared earlier research [32] using the hierarchical clustering (agglomerative method), where we were able to forecast the ideal number of clusters by using the dendrogram result shown in Figure 4. Although we were able to anticipate from the dendrogram diagram that the longest vertical line free of any interference will be considered as the ideal cluster in this case of hierarchical clustering, we did not provide any predetermined cluster values to the algorithm. In this Figure 4, either 2 or 4 can be stated, but we have opted to choose 4 as the ideal number of clusters for further processing in order to obtain the most interesting facts.

The elbow approach, grouping from the k-means algorithm [31], and the dendrogram of the hierarchical clustering algorithm [32] all produced results that divided the clustered sequences of these mushroom sequences into four groups. The best cluster size, as determined by the research, is 4. The algorithm for affinity propagation produced the same outcomes. Additionally, because the affinity algorithm does not require us to supply any beginning parameters as a cluster value, that provides confidence that the suggested technique is on track to group the current mushroom sequences. The complete dataset is trained by the algorithm, which then generates the result.

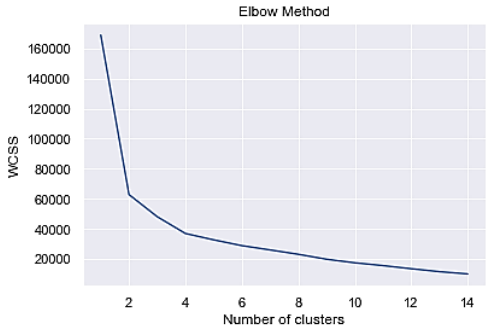


Fig 3. Elbow method using KMeans attributes to measure number of clusters [31]

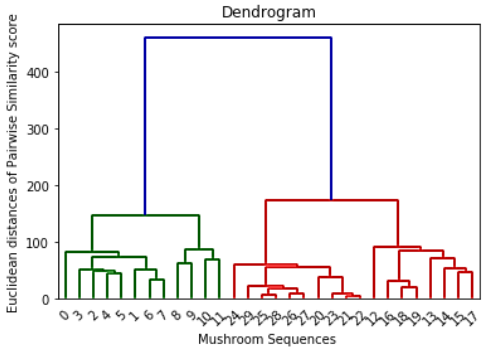


Fig 4. Dendrogram diagram to find the optimal number ofclusters using hierarchical clustering [32]

Table 6 compares the duration and complexity of these three algorithms and shows that the affinity propagation method performs better than KMeans and hierarchical algorithms in terms of time and space complexity constraints. All three methods allocate space similarly when viewed from the perspective of space complexity. There isn't much of a distinction between them. As a result, we are unable to draw any firm conclusions; yet, affinity propagation has proven to be more successful than the hierarchical clustering approach.

Table 6. Time and Space complexity

Time complexity	Space complexity
KMeans – 24.35 ms	KMeans- 140.95 MiB
Hierarchical clustering – 21.7 ms	Hierarchical Clustering – 150.23 MiB
Affinity Propagation – 13.7 ms	Affinity Propagation – 143.48 MiB

5.1 Cluster validation metrics results and discussion

Every research project should include quality assurance testing to ensure the accuracy of their findings. We have combined many clustering methods into our suggested technique, therefore testing our clustering algorithm from the standpoint of validating the cluster number where the groups are to be produced is unquestionably a necessary step. Since the unsupervised clustering model in the current study does not include any predetermined labels to address the division of cluster groups, the validation metrics were constructed utilising the Silhouette score, Calinski-Harabasz Index, and Davies-Bouldin Index approaches. These measures' primary function is to forecast the ideal number of clusters for each of the methods described in this article. We have chosen 2,3,4,5,6,7, and 8 as the cluster numbers in order. The entire validation procedure operates in a loop, according to this cluster number, to get the metrics for all of the clusters and determine which one is the best. In the Figure 5, We used the silhouette score to examine, and since the result should range from -1 to 1, we should choose the highest score that is close to 1 as the best. Figure 5 data showed that the values of cluster numbers 2 to 5 of k-means are identical to those of the hierarchical method, while the values are somewhat different for cluster numbers 6, 7, and 8. According to the validation of the silhouette score, we should concentrate on the nearest value of 1, in Figure 5. The top score chooses the number of clusters 2, 3, and 4, while the affinity propagation algorithm hits cluster 4 with a value of 0.351, which is same for both algorithms. In a similar manner, while examining Figure 6, we should use the Calinski-Harabasz Index and choose the highest score for the specified number of clusters. The affinity propagation algorithm likewise gave a result of 30.907, which indicates the same cluster number value for 4 in both techniques, for the cluster number 2,3,4 as in the topmost algorithm. We used the Davies-Bouldin Index in Figure 7 to forecast the number of clusters by indicating that the outcome value should be as low as possible. At that moment, all three methods have an equal number of clusters. For the k-means and hierarchical algorithms, the lowest point sequentially designates cluster numbers 2, 3, and 4, whereas the affinity propagation technique yields a value of 1.143 for cluster number 4. According to the aforementioned metrics, we can infer that the ideal number of clusters can range from 2 to 4, but given the elbow method, dendrogram, and affinity propagation algorithm's results, which are independent of the working models' cluster counts, we can infer that the ideal number of clusters is 4 for this application.

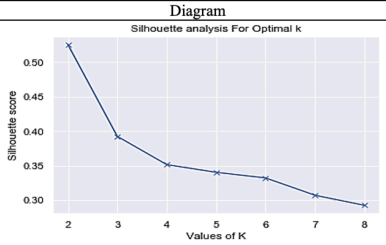
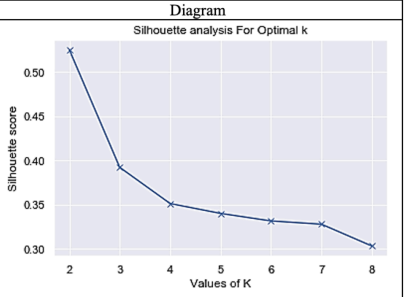
K-Means algorithm	Diagram
silhouette score for 2 no. of clusters 0.525 silhouette score for 3 no. of clusters 0.392 silhouette score for 4 no. of clusters 0.351 silhouette score for 5 no. of clusters 0.340 silhouette score for 6 no. of clusters 0.332 silhouette score for 7 no. of clusters 0.307 silhouette score for 8 no. of clusters 0.292	
Hierarchical algorithm	**Diagram**
silhouette score for 2 no. of clusters 0.525 silhouette score for 3 no. of clusters 0.392 silhouette score for 4 no. of clusters 0.351 silhouette score for 5 no. of clusters 0.340 silhouette score for 6 no. of clusters 0.331 silhouette score for 7 no. of clusters 0.328 silhouette score for 8 no. of clusters 0.303	
Affinity Propagation algorithm - Silhouette Score: 0.351	

Fig 5. Silhouette Score for K-Means, Hierarchical, Affinity propagation algorithm

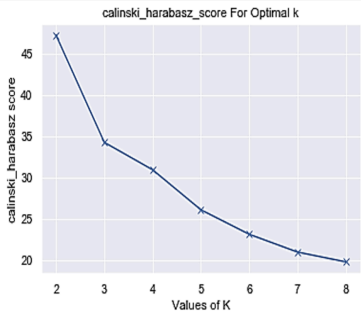
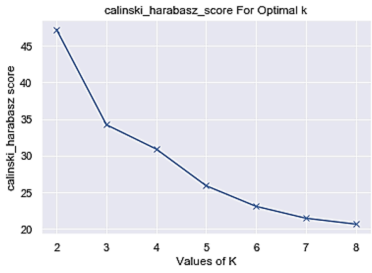
K-Means	Diagram
calinski_harabasz_score 2 no. of clusters 47.210 calinski_harabasz_score 3 no. of clusters 34.227 calinski_harabasz_score 4 no. of clusters 30.907 calinski_harabasz_score 5 no. of clusters 26.061 calinski_harabasz_score 6 no. of clusters 23.096 calinski_harabasz_score 7 no. of clusters 20.941 calinski_harabasz_score 8 no. of clusters 19.753	
Hierarchical algorithm	**Diagram**
calinski_harabasz_score 2 no. of clusters 47.210 calinski_harabasz_score 3 no. of clusters 34.227 calinski_harabasz_score 4 no. of clusters 30.907 calinski_harabasz_score 5 no. of clusters 25.908 calinski_harabasz_score 6 no. of clusters 23.096 calinski_harabasz_score 7 no. of clusters 21.471 calinski_harabasz_score 8 no. of clusters 20.670	
Affinity propagation algorithm calinski_harabasz_score 30.907	

Fig 6. Calinski-Harabasz Index score for K-Means, Hierarchical, Affinity propagation algorithm

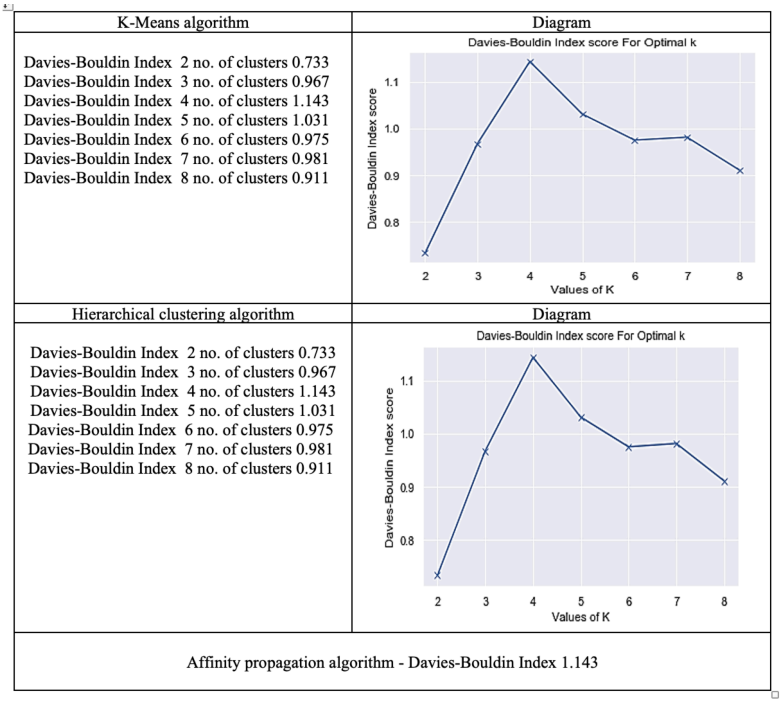


Fig 7. Davies-Bouldin Index score for K-Means, Hierarchical, Affinity propagation algorithm

6 Conclusion

The investigation into the clustering of mushroom 5.8s rRNA sequences has been conducted using an affinity propagation technique that yields four cluster groups as findings, which are also obtained using the k-means, hierarchical clustering approach (the work has already been published). The efficiency of identifying the optimal number of clustering values is also worked well using the elbow method, dendrogram, and in the automated results of the affinity propagation algorithm. Finally, we have four groups of clustered mushroom sequences that might be like each other in their own biological nature. The findings may aid biotechnology researchers in their search for additional evidence on these grouping sequences before beginning laboratory work. The originality behind this research work is that we cannot explore it in the cd-hit tool, which is a frequently accessible clustering tool for biological sequences to measure. The numerical statistical value will help them to move on confidently by working towards these clustered mushroom sequences for the reality of their own biological similarity to each other. From the perspective of information technology, the affinity propagation algorithm has better time and space complexity than the k-means and hierarchical clustering algorithms. The article truly demonstrates a novel approach towards clustering the sequences using the PIM matrix and delivers the implementation of retrieving the optimal value of clustering for any dataset by conquering the results using cluster validation metrics like silhouette score, calinski harabasz score, and Davies-Bouldin Index, which propels the results to ensure the optimal number of clusters. By identifying the sustainability of clustering aspects on any given dataset, this proposed method will help researchers for better understanding of the work flow.

References

1) Rajoub B. Chapter 3 - Supervised and unsupervised learning. *Biomedical Signal Processing and Artificial Intelligence in Healthcare*. 2020;p. 51–89. Available from: https://doi.org/10.1016/B978-0-12-818946-7.00003-2.
2) Kaur N, Virk U, Kumari. Genome Sequence Analysis of Lungs Cancer Protein WDR74 (WD Repeat-Containing Protein). *International Journal for Research in Applied Science & Engineering Technology*. 2022;10(5).
3) Venegas CN. Identification of genomes: Clustal Omega and BLAST: One introduction. *International Journal of Science and Research*. 2022;6(2):26–29. Available from: https://doi.org/10.30574/ijsra.2022.6.2.0154.
4) Katoh K. Multiple Sequence Alignment. 2021;p. 1–321. Available from: https://doi.org/10.1007/978-1-0716-1036-7.
5) Asraf SS, Sivakkanni A, Sneha M, Janani S, Jashin P, Jemimal AM. In Silico Based Bioinformatics Project During the COVID-19 Lockdown Period: An Alternative to Wet Lab Study. *Journal of Engineering Education Transformations*. 2022;35(3):82–87. Available from: https://dpo.org/10.16920/jeet/2022/v35i3/22090.
6) . . Available from: http://weizhong-lab.ucsd.edu/cd-hit/.

7) Roknabadi S, Sadatabdosalehi A, Pouyamehr F, Koohi S. An accurate alignment-free protein sequence comparator based on physicochemical properties of amino acids. *Scientifc Reports 2022*. 2022;12:11158. Available from: https://doi.org/10.1038/s41598-022-15266-8.
8) Gonçalves RS, Musen MA. The variable quality of metadata about biological samples used in biomedical experiments. *Scientific Data*. 2019;6(1):190021. Available from: https://doi.org/10.1038/sdata.2019.21.
9) Laureano LB, Sison AM, Medina RP. Handling Imbalanced Data through Affinity Propagation and SMOTE. ICCBD TAICHUNG, Taiwan Association for Computing Machinery. *ACM* . 2019;p. 22–26. Available from: https://doi.org/10.1145/3366650.3366665.
10) Emami N, Pakzad A. A New Knowledge-based System for Diagnosis of Breast Cancer by a combination of Affinity Propagation Clustering and Firefly Algorithm. *Journal of AI and Data Mining*. 2019;7(1):59–68. Available from: https://doi.org/10.22044/JADM.2018.6489.1763.
11) Wang Y, Peng Q, Pei Z, Ma M, Chen Y, Leng C, et al. Detection of Social Groups in Class by Affinity Propagation. *2019 IEEE International Conferences on Ubiquitous Computing & Communications (IUCC) and Data Science and Computational Intelligence (DSCI) and Smart Computing, Networking and Services (SmartCNS)*. 2019;p. 484–489. Available from: https://doi.org/10.1109/IUCC/DSCI/SmartCNS.2019.00106.
12) Taheri S, Bouyer A. Community Detection in Social Networks Using Affinity Propagation with Adaptive Similarity Matrix. *Big Data*. 2020;8(3):189–202. Available from: https://doi.org/10.1089/big.2019.0143.
13) Marin FR, Dávalos A, Kiltschewskij D, Crespo MC, Cairns M, Andrés-León E, et al. RNA-Seq, Bioinformatic Identification of Potential MicroRNA-like Small RNAs in the Edible Mushroom Agaricus bisporus and Experimental Approach for Their Validation. *International Journal of Molecular Sciences*;23(9):4923–4923. Available from: https://doi.org/10.3390/ijms23094923.
14) Wang X, Xu Y. An improved index for clustering validation based on Silhouette index and Calinski-Harabasz index. *IOP Conference Series: Materials Science and Engineering*. 2019;569(5):052024. Available from: https://doi.org/10.1088/1757-899X/569/5/052024.
15) Wu Z, Zhang Y, Zhang JZ, Xia K, Xia F. Determining Optimal Coarse-Grained Representation for Biomolecules Using Internal Cluster Validation Indexes. *Journal of Computational Chemistry*. 2019;41(1):14–20. Available from: https://doi.org/0.1002/jcc.26070.
16) Simić S, Villar JR, Calvo-Rolle JL, Sekulić SR, Simić SD, Simić D. An Application of a Hybrid Intelligent System for Diagnosing Primary Headaches. *International Journal of Environmental Research and Public Health*. 1890;18(4):1890. Available from: https://doi.org/10.3390/ijerph18041890.
17) Ünlü R, Xanthopoulos P. Estimating the number of clusters in a dataset via consensus clustering. *Expert Systems with Applications*. 2019;125:33–39. Available from: https://doi.org/10.1016/j.eswa.2019.01.074.
18) Anitha S, Metilda DM. An Extensive Investigation Of Outlier Detection By Cluster Validation Indices. 2019. Available from: http://dx.doi.org/10.13140/RG.2.2.26801.63848. doi:http://dx.doi.org/10.13140/RG.2.2.26801.63848.
19) Wang W, Ma Q, Liu Y, Yao N, Liu J, Wang Z, et al. Clustering analysis method of power grid company based on K-means. *Journal of Physics: Conference Series*. 2021;1883(1):012072. Available from: https://doi.org/10.1088/1742-6596/1883/1/012072.
20) Morales F, García-Torres M, Velázquez G, Daumas-Ladouce F, Gardel-Sotomayor PE, Vela G, et al. Analysis of Electric Energy Consumption Profiles Using a Machine Learning Approach: A Paraguayan Case Study. 2022;11(2):267. Available from: https://doi.org/10.3390/electronics11020267.
21) Mughnyanti M, Efendi S, Zarlis M. Analysis of determining centroid clustering x-means algorithm with davies-bouldin index evaluation. *IOP Conference Series: Materials Science and Engineering*. 2020;725(1):012128. Available from: https://doi.org/10.1088/1757-899X/725/1/012128.
22) Wijaya DAYA, Kurniady E, Setyanto W, Tarihoran DS, Rusmana R, Rahim. Davies Bouldin Index Algorithm for Optimizing Clustering Case Studies Mapping School Facilities. *TEM Journal 2021*;3(0):1099–1103.
23) Ramadhani S, Azzahra D, Tomi Z. Comparison of K-Means and K-Medoids Algorithms in Text, Mining based on Davies Bouldin Index Testing for Classification of Student's Thesis. *Jurnal Teknologi Informasi dan Komunikasi*. 2022;13(1). Available from: https://doi.org/10.31849/digitalzone.v13i1.
24) Punhani R, Arora VPS, Sabitha AS, Shukla VK. Segmenting e-Commerce Customer through Data Mining Techniques. *Journal of Physics: Conference Series*. 2021;1714(1). Available from: https://doi.org/0.1088/1742-6596/1714/1/012026.
25) Sari PK, Purwadinata A. Analysis Characteristics of Car Sales In E-Commerce Data Using Clustering Model. *Journal of Science and Its Application*. 2019;2(1):19–28. Available from: https://doi.org/10.21108/jdsa.2019.2.19.
26) Umam MWF, Fatekurohman M, Anggraeni D. Hybrid clustering and classification methods to find out the pattern of the spread of covid-19 in East Java province. *Journal of Physics: Conference Series*. 2022;2157(1):012030. Available from: https://doi.org/10.1088/1742-6596/2157/1/012030.
27) Shahapure CKR, Nicholas. Cluster Quality Analysis Using Silhouette Score. *IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA)*. 2020;2020. Available from: https://doi.org/10.1109/DSAA49011.2020.00096.
28) Naghizadeh A, Metaxas DN. Condensed Silhouette: An Optimized Filtering Process for Cluster Selection in K-Means. *Procedia Computer Science*. 2020;176(176):205–214. Available from: https://doi.org/10.1016/j.procs.2020.08.022.
29) Sundari TM, Anand AAP, Jenifer P, Shenbagarathai R. Bioprospection of Basidiomycetes and Molecular Phylogenetic Analysis Using Internal Transcribed Spacer (ITS) and 5.8S rRNA Gene Sequence. *Scientific Reports*. 2018;8(10720). Available from: https://doi.org/10.1038/s41598-018-29046-w.
30) Sudhasini P, Ashadevi B. Pairwise Sequence Alignment Similarity Score Prediction on Mushroom Biological data. *International Journal of Advanced Science and Technology*. 2020;29(4s):1844–1867. Available from: http://sersc.org/journals/index.php/IJAST/article/view/6993.
31) Sudhasini P, Ashadevi B. Clustering Mushroom 5.8s rRNA Sequences using k-means Algorithm with Predicted k Value. In: 5th international conference on intelligent computing and control systems, IEEE. 2021. Available from: https://doi.org/10.1109/ICICCS51141.2021.9432167.
32) Masoodi F, Quasim M, Bukhari S, Dixit S, Alam S. Applications of Machine Learning and Deep Learning on Biological Data. 1st ed. and others, editor;Auerbach Publications. Taylor & Francis. CRC press. 2023. Available from: https://www.routledge.com/Advances-in-Computational-Collective-intelligence/book-series/ACCICRC.