DOI 10.15276/imms.v13.no1-2.5
UDC 004.043

Informatics and Mathematical Methods in Simulation
Vol. 13 (2023), No. 1-2, pp. 5-15

HIERARCHICAL CLUSTERING ALGORITHM FOR DENDROGRAM CONSTRUCTION AND CLUSTER COUNTING

N.I. Boyko, O.A. Tkachyk

Lviv Polytechnic National University, Knyaz Roman Str. 5 Lviv, 79013; Ukraine; nataliya.i.boyko@lpnu.ua; oleksandr.a.tkachyk@lpnu.ua

The article provides a comprehensive overview of hierarchical clustering and dendrogram construction, with a focus on the methods used for determining the optimal number of clusters. The article discusses the theoretical foundations of hierarchical clustering and the process of constructing dendrograms, and goes on to describe several popular methods for determining the number of clusters. The article focused on both divisive and agglomerative clustering methods and the dendrogram, the advantages and disadvantages of each method, and how dendrograms are used to visualize the results of hierarchical clustering. It also provides comparison of hierarchical clustering with non-hierarchical clustering, particularly the K-means algorithm, and discusses their respective advantages and disadvantages. One of the key advantages of hierarchical clustering is that it does not require the user to specify the number of clusters in advance, as is the case with non-hierarchical clustering. Instead, a dendrogram can be used to determine the appropriate number of clusters. The article concludes by noting the usefulness of hierarchical clustering for a range of applications, particularly in exploratory data analysis. The article also covers the main methods to identify which objects and clusters are most similar. Additionally, the article provides an overview of the K-means clustering method and compares it to hierarchical clustering.

Keywords: agglomerative clustering, divisive clustering, hierarchical clustering, k-means.

Introduction. Cluster analysis has proven to be an invaluable tool in the context of multidimensional datasets and its utility in research and uncontrolled analysis. It highlights the hierarchical approach to clustering as a popular method in genomics and other fields due to its ability to reveal multiple layers of clustering structure simultaneously. This allows researchers to gain a deeper understanding of the data, identify patterns, and develop hypotheses or inform decision-making. Many applied problems, measuring the degree of similarity between objects is often much simpler than forming descriptive features. For example, taking two photos and immediately identifying that they both depict the same person is much easier than understanding the specific features that make them similar. The task of object classification based on their similarity, without any predefined classes to which the objects can be assigned, is called clustering [2, 15].

Hierarchical clustering is a popular technique used for data analysis and pattern recognition. It is a method that groups data objects based on their similarities and differences. This technique can be used to explore the underlying structure of a dataset, identify relationships between variables, and discover patterns in data [12, 20].

Hierarchical clustering algorithms can be divided into two main types: divisive and agglomerative. Divisive clustering starts with a single cluster containing all data objects and divides it into smaller clusters until each object is in its own cluster [1,3]. On the other hand, agglomerative clustering starts with each object in its own cluster and merges them together until all objects belong to a single cluster [7]. One of the most important outputs of hierarchical clustering is the dendrogram. A dendrogram is a tree-like diagram that illustrates the hierarchical relationships between clusters. It displays the order in which clusters are merged and the distances between them. This diagram helps to visualize the hierarchical structure of the data and provides insights into the relationships between clusters [5, 11].

Content statement of the problem. The purpose of study is to apply different algorithms for constructing dendrograms and determining the optimal number of clusters in hierarchical clustering. The study aims to provide a theoretical background of hierarchical clustering, explain the process of dendrogram construction, and discuss the different methods for determining the number of clusters.

The object of this study is to analyze different clustering algorithms for constructing dendrograms and determining the number of clusters in hierarchical clustering. The study focuses on the methodology and practical aspects of hierarchical clustering, including different types of linkage methods, distance metrics, initialization of clusters, and scalability issues. However, the study also compares several non-hierarchical clustering algorithms such as k-means. Therefore, the object of the study includes both hierarchical and non-hierarchical clustering algorithms, and aims to provide a comprehensive and practical guide to different clustering methods for researchers and practitioners in the field. The study aims to provide guidance and insights into the process of dendrogram construction and the determination of the optimal number of clusters. The object of the study is the algorithmic approach to hierarchical clustering and its applications in data analysis and research.

Analysis of recent resources. The paper [1] by M. Kuchaki Rafsanjani is a research paper that provides an extensive overview of the various hierarchical clustering algorithms. The paper discusses the different types of hierarchical clustering, such as agglomerative and divisive clustering, and the pros and cons of each type. The paper includes a comparative analysis of several hierarchical clustering algorithms, including Ward's method, k-means clustering, and spectral clustering, and provides insights into the strengths and weaknesses of each algorithm. The author concludes the paper by discussing some of the current research directions in hierarchical clustering, such as semi-supervised clustering and graph-based clustering, and highlighting the potential applications of hierarchical clustering in various fields, including bioinformatics and image analysis [10, 13].

In the research [2, 17] Luben M. C. Cabezas discusses different approaches to dendrogram construction and visualization. Also, the author explores the use of dendrograms for interpreting machine learning models and extracting insights from large datasets.

Research [3, 19] is interesting by its discussion of different methods for determining the number of clusters in hierarchical clustering, including the silhouette method and the elbow method. The paper also covers other important aspects of k-means clustering, such as the choice of distance metrics and initialization of clusters, and provides insights into the strengths and weaknesses of different methods.

The article [4, 20] provides a survey of agglomerative clustering algorithms and their applications in high-dimensional data. Authors describe and present a comprehensive classification of different clustering techniques for high dimensional data. The article covers other important aspects of clustering validation, such as the choice of distance metrics and the interpretation of validation results. Furthermore, the paper presents experimental results and case studies to illustrate the effectiveness and limitations of different clustering validation measures.

Methods and tools of research. The approach of creating a dendrogram, which is a type of tree diagram that shows the arrangement of clusters produced by a clustering algorithm, using specific algorithms that are intended for this purpose. These algorithms take the data input and produce a hierarchical structure of clusters that can be visualized in the form of a dendrogram [6, 8].

Hierarchical clustering is a type of cluster analysis. One of the great advantages of hierarchical algorithms over non-hierarchical ones is their hierarchical structure, which is created during the algorithm's operation. The operation of such an algorithm can be represented as a dendrogram (Fig. 1) or, as they are also called, a tree diagram. In the dendrogram, each level corresponds to one iteration of the algorithm [9, 14].

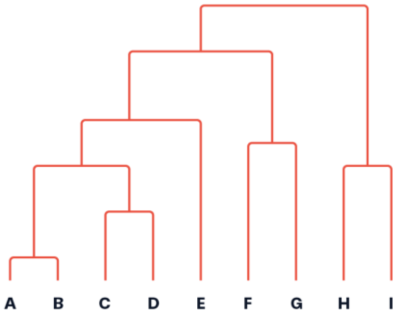

Fig. 1. Dendrogram example

The number of clusters can either decrease or increase with each iteration, depending on the type of hierarchical clustering algorithm used.

Types of hierarchical clustering. As already mentioned, there are two main types of hierarchical clustering. Each of them moves in a different direction. These are agglomerative and divisive algorithms. In the first case, we start with each object of the study having its own cluster, and with each iteration, clusters begin to merge until all objects end up in one cluster. With the divisive algorithm, on the other hand, everything is the opposite. Initially, we have one large cluster that includes all objects of our study, and with each iteration, we begin to divide this cluster into smaller ones, which ultimately leads to each object having only its own cluster [16, 18]. Approximate working of both algorithms is shown in Figure 2.

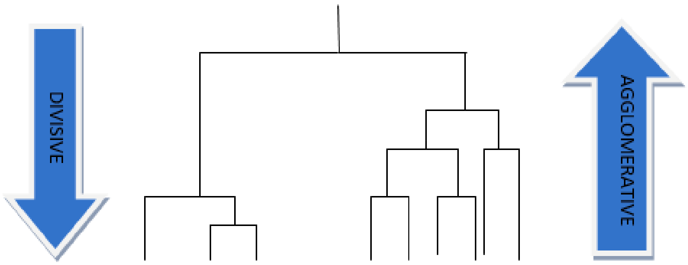

Fig. 2. Agglomerative and divisive algorithms

Agglomerative hierarchical clustering. Let's take a closer look at the agglomerative method. As a dataset, we will take several coordinates X and Y described in table 1 to demonstrate the operation of the agglomerative method.

Table 1

The objects under study

№	1	2	3	4	5	6	7	8	9	10	11	12	13
x	6	4.9	8.2	7.1	2	1	1.5	2.8	3	6.9	6.1	8	7.1
y	0.6	3	2.1	3.8	6	7.8	8.3	7	7.9	6.9	8.2	7.9	8.8

First, we have our objects of study, which are shown in Figure 3.

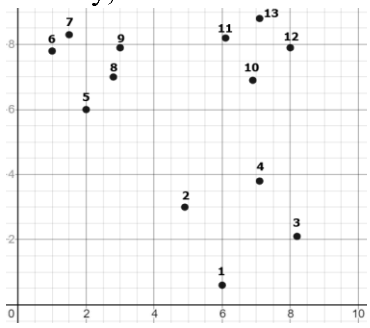

Fig. 3. Study objects represented on the coordinate plane

At the beginning of the algorithm, all of our objects form their own clusters (Fig. 4).

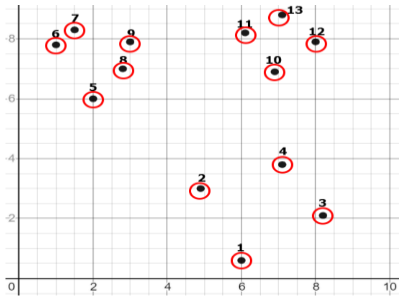

Fig. 4. Initial clusters

Now each point is in its own cluster. The next step is to merge clusters with the smallest distance between them. In our case, these turned out to be clusters 6 and 7, as their Euclidean distance for X and Y are:

$$d = \sqrt{(1.5-1)^2 + (8.3-7.8)^2} = \sqrt{0.5} \approx 0.7071 \quad (1)$$

Now we have a new cluster number 14 (Fig. 5).

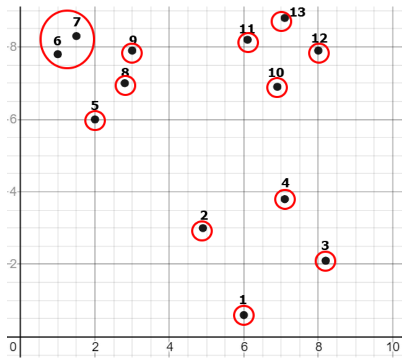

Fig. 5. Points 6 and 7 forms a new cluster

Continuing to search for clusters with the smallest distances, we find that clusters 8 and 9, 11 and 13, and 10 and 12 have the smallest distances, and they form new clusters. Data is provided in table 2 and visualized on Figure 6.

Table 2

Cluster combination

Clusters A & B	Cluster A coordinates	Cluster B coordinates	Distance between clusters	New cluster number
8 and 9	{2.8, 7}	{3, 7.9}	0.9219	15
11 and 13	{6.1,8.2}	{7.1,8.8}	1.1661	16
10 and 12	{6.9,6.9}	{8,7.9}	1.4866	17

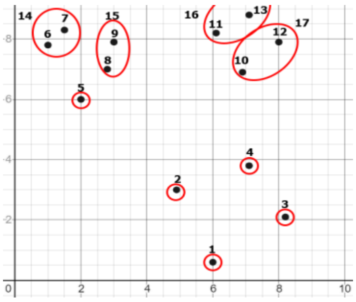

Fig. 6. New cluster formation

At this stage, our dendrogram looks as on Figure 7.

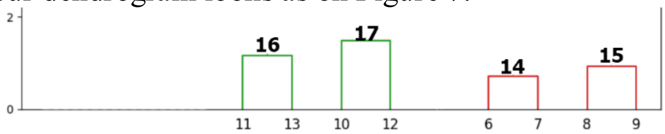

Fig. 7. Updated dendrogram based on recent updates

Now we can see that the clusters closest to each other are not single-point clusters, but a cluster with multiple objects and a single-point cluster. In this case, we need a measure to compute the distances between clusters with multiple points. We will continue merging clusters using Complete Linkage. The results are seen in Table 3 and Figure 8.

Table 3

Cluster combination

Clusters A & B	Distance between clusters	New cluster number
16 and 17	1.9235	18
14 and 15	2.0024	19

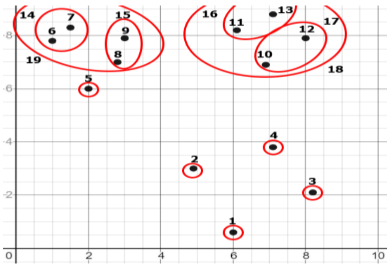

Fig. 8. Newly formed clusters

Continuing this algorithm, we will eventually end up with all objects in one cluster. Let's try to finish the work using Python. After performing agglomerative clustering using the Complete Linkage method, which we will discuss later, I obtained the following data array displayed on Figure 9.

```
array([[ 6,  7,  0.70710678, 2, 14],
       [ 8,  9,  0.92195445, 2, 15],
       [11, 13,  1.16619038, 2, 16],
       [10, 12,  1.48660687, 2, 17],
       [16, 17,  1.92353841, 4, 18],
       [14, 15,  2.00249844, 4, 19],
       [ 3,  4,  2.02484567, 2, 20],
       [ 5, 19,  2.35372046, 5, 21],
       [ 1,  2,  2.64007576, 2, 22],
       [20, 22,  3.42052628, 4, 23],
       [18, 21,  7.00071425, 9, 24],
       [23, 24,  9.18313672, 13, 25]])
```

Fig. 9. The array obtained using Python

The first two columns represent the cluster numbers that will be merged. The third column represents the distance between them, the fourth column indicates the number of objects that will be in the new cluster, and the last column represents the number of the newly created cluster. Indeed, if we look at this table, its first half completely matches the data that we calculated above (Fig. 10).

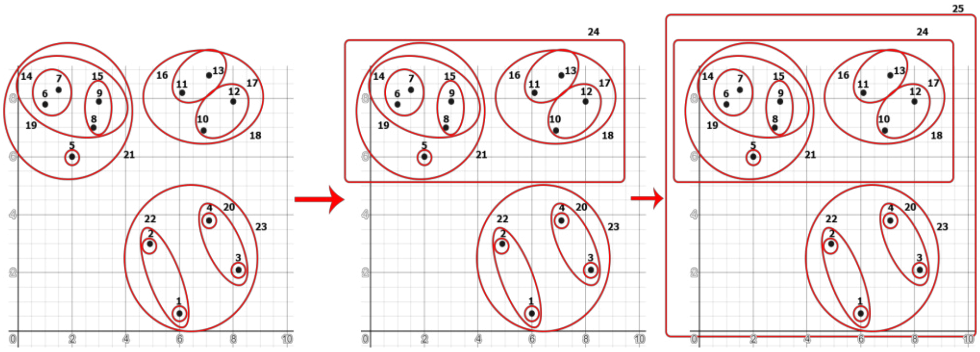

Fig. 10. The last iterations of the agglomerative hierarchical clustering method

Now all research objects are in the same cluster with number 25. It may seem illogical as it does not give us any meaningful data. However, in fact, it is quite the opposite. Since the algorithm is hierarchical, we can obtain data that was obtained at a certain iteration. This can be visualized more clearly by looking at the result displayed in the dendrogram (Fig. 11).

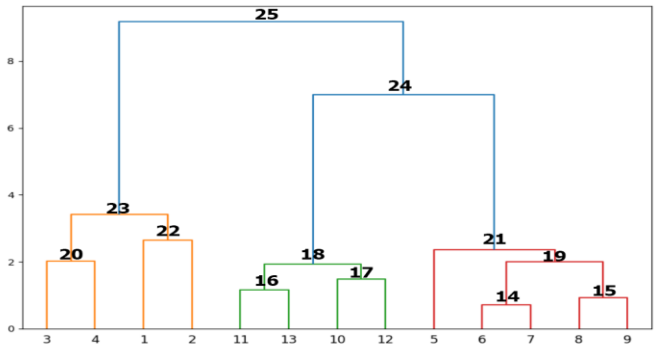

Fig. 11. Newly formed dendrogram based on coordinates

Now we have obtained a dendrogram that shows all the iterations of creating new clusters. Our dendrogram also shows that indeed points 6 and 7 were the first to merge, then 8 and 9, and 11 and 13. There are also three strongly pronounced clusters 23, 18, and 21. Let's visualize them on a plot by coloring the points belonging to these clusters in different colors (Fig. 12).

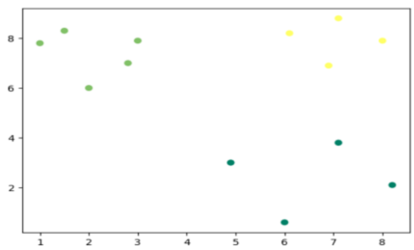

Fig. 12. The three clusters obtained using the agglomerative method

We can also notice that cluster 23 is much taller than the other two. This is due to the fact that the distances between points in this cluster are much larger, making the cluster itself larger in size (Fig. 13).

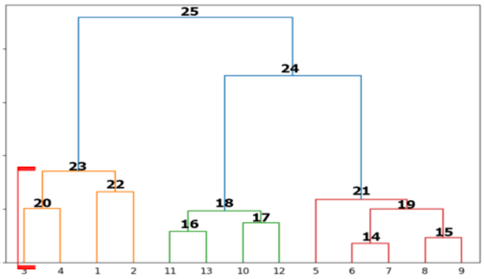

Fig. 13. The height of cluster 23

Divisive clustering method. In divisive or divisive clustering, everything happens the opposite way. First, we have a cluster that contains all the objects we are studying (Fig. 14).

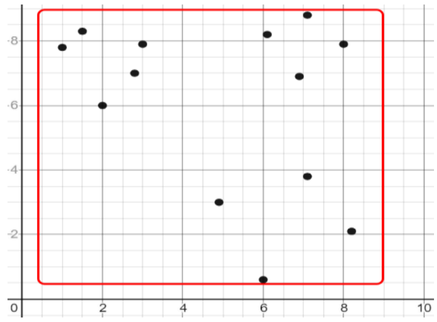

Fig. **14**. Beginning of divisive clustering algorithm method

Immediately after the first iteration, our large cluster is divided into smaller clusters, and those in turn into even smaller ones. This process continues until all objects end up in their own clusters (Fig. 15).

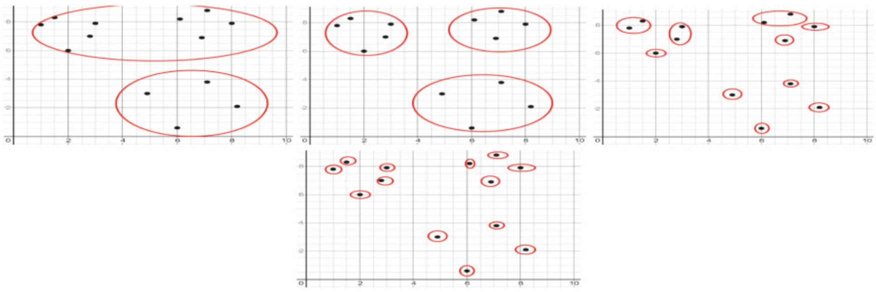

Fig. **15**. Results of divisive clustering algorithm method

The recalculation of distance between clusters. To determine which clusters to merge, we need a metric that will measure the similarity between clusters. There are five main methods to identify which objects and clusters are most similar.

Single Linkage is a method that begins by finding the two closest objects that form the primary cluster. Each subsequent object is then added to the cluster that is closest to one of its objects (Formula 2).

$$d_{min}(C_i, C_j) = min_{x_i \in C_i, x_{i,} \in C_i} d(x_i, x_j) \quad (2)$$

Complete Linkage, also known as the maximum linkage method, is the inverse of Single Linkage. The rule for combining clusters in this method is based on finding the two objects that are furthest apart from each other (Formula 3).

$$d_{max}(C_i, C_j) = max_{x_i \in C_i, x_{i,} \in C_i} d(x_i, x_j) \quad (3)$$

Advantage Linkage – at each step, the average distance between each object from one cluster and each object from another cluster is calculated. An object is assigned to a given cluster if the average distance is smaller than the average distance to any other cluster (Formula 4).

$$d_{avg}(C_i, C_j) = \frac{1}{|C_i||C_j|} \sum_{x_i \in C_i} \sum_{x_i \in C_i} d(x_i, x_j) \tag{4}$$

Ward's Method: this method minimizes the variance between all clusters by selecting clusters that result in the smallest increase in overall variance [8].

Centroid Method: this method calculates the distance between the centroids of each cluster.

Comparison of hierarchical and non-hierarchical clustering. Before moving on to the topic of cluster count, it would be helpful to understand other clustering methods. Non-hierarchical clustering also has many methods and algorithms, but we will only discuss the K-means algorithm (nearest neighbor method).

Let's suppose there are hypotheses about the number of clusters, let's say N clusters. In this case, the program can be set to N clusters. This is precisely the use case for the K-means method. While in hierarchical clustering, we can choose the number of clusters after the program has finished processing the data, in non-hierarchical clustering, specifically in the nearest neighbor method, we must determine the number of clusters in advance. This is a major drawback of the algorithm because it is not always possible to know or guess how many clusters there may be after processing the data.

In general, K-means is a popular clustering algorithm that has several advantages:

1. Simplicity: The algorithm is easy to understand and implement, making it a popular choice for data analysts and scientists.
2. Scalability: K-means is a relatively fast and efficient algorithm that can handle large datasets.
3. Ability to handle continuous variables: K-means works well with continuous variables, such as age or income, as opposed to categorical variables.
4. Reproducibility: The results of K-means are reproducible, meaning that if you run the algorithm multiple times with the same inputs, you should get the same results every time.

But also, K-means algorithm has its own disadvantages which is provided in the list below:

1. Requires the number of clusters to be specified: One of the main disadvantages of K-means is that it requires the number of clusters to be specified beforehand. This can be a major drawback, especially when the data does not naturally lend itself to a specific number of clusters.
2. Sensitive to initial cluster centers: The final clusters produced by K-means can be highly dependent on the initial random selection of cluster centers. This can lead to suboptimal results if the initial centers are not representative of the data
3. Outliers can heavily influence results: K-means is highly sensitive to outliers in the data. Outliers can heavily influence the position of the cluster centers and lead to suboptimal clustering results.
4. Cannot handle non-linear data: K-means algorithm assumes that the data can be separated into clusters based on linear boundaries. Therefore, it may not perform well on non-linear data.

Result and discussion. One of the main advantages of hierarchical clustering is that it does not require prior knowledge of the number of clusters. At the end of the process, we obtain a hierarchy of clusters, which is usually represented as a dendrogram. By analyzing the dendrogram, we can determine the number of clusters. Additionally, the number of clusters depends on how distinct we want the differences between objects in the cluster to be. If more detail is required, then the dendrogram of the research can be truncated somewhere around the 1 mark. If less differentiation between objects in one cluster is needed or a smaller number of clusters is desired, then the dendrogram can be truncated at the 2.5 mark (Fig. 16).

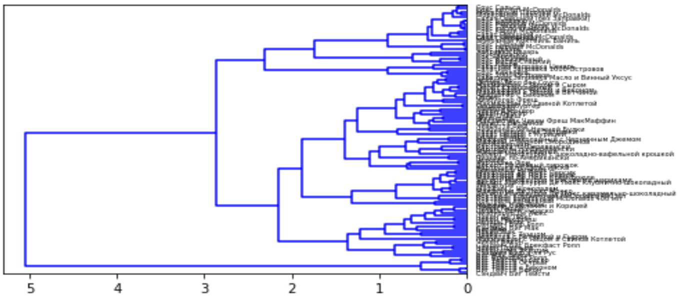

Fig. 16. Determining the optimal number of clusters

Conclusion. Hierarchical clustering is a popular method for grouping objects into clusters based on their similarity. It involves creating a hierarchy of nested clusters, represented as a dendrogram, which can be used to explore relationships among the objects. There are two main types of hierarchical clustering: agglomerative and divisive. Agglomerative clustering involves starting with each object in its own cluster and iteratively merging the closest clusters until all objects belong to a single cluster. In contrast, divisive clustering starts with all objects in a single cluster and recursively splits them into smaller clusters. While hierarchical clustering does not require prior knowledge of the number of clusters, it can be computationally expensive and may be less effective than other clustering methods for large datasets. K-means clustering, a non-hierarchical method, is often used instead as it requires a prior specification of the number of clusters and is more computationally efficient. However, hierarchical clustering has some unique advantages. The dendrogram can provide insights into the structure of the data and can be used to determine the number of clusters based on the desired level of similarity among objects. Additionally, agglomerative clustering allows for exploration of the hierarchy of clusters, which can be useful in identifying subgroups and relationships among objects. As a conclusion, the choice of clustering method depends on the specific characteristics of the data and the research question. Hierarchical clustering can be a powerful tool for exploratory analysis and uncovering patterns in data.

References

1. Kuchaki Rafsanjani M., Asghari Varzaneh Z., Emami Chukanlo N. A Survey of Hierarchical Clustering Algorithms. *TJMCS*. 2012. Vol. 5. No. 3. pp. 229-240. DOI: 10.22436/jmcs.05.03.11
2. Luben M. C. Cabezas, Izbicki R., Stern R. B. Hierarchical clustering: visualization, feature importance and model selection. 2023. URL: https://arxiv.org/pdf/2112.01372.pdf
3. Rathiga P., Selvi P. To determine the optimal number clusters. *Journal of Emerging Technologies and Innovative Research (JETIR)*. 2020. Vol.7, Issue 10. pp. 167-172.
4. Pavithra M.S., Parvathi R. M. A Survey on Clustering High Dimensional Data Techniques. *International Journal of Applied Engineering Research*. 2017. Vol. 12, No. 11. pp. 2893-2899.
5. Ward Jr. Hierarchical grouping to optimize an objective function. *Journal of the American statistical association*. 1963. Vol. 58(301). pp. 236–244.
6. Timofeeva A. Evaluating the robustness of goodness-of-fit measures for hierarchical clustering. *Journal of Physics: Conference Series*. 2019. Vol. 1145. pp. 012049.
7. Roux M. A Comparative Study of Divisive and Agglomerative Hierarchical Clustering Algorithms. *Journal of Classification*. 2018. Vol. 35 (2). pp. 345-366. ff10.1007/s00357-018-9259-9ff. ffhal02085844.

8. Murtagh F., Legendre P. Ward's Hierarchical Agglomerative Method: Which Algorithms Implement Ward's Criterion? *Journal of Classification*. 2014. Vol. 31. pp. 274–295.
9. Sarker A., Shamim S.M., Shahiduz Zama Dr. Md., Mustafizur Rahman Md. Employee's performance analysis and prediction using K-means clustering & decision tree algorithm. *Global Journal of Computer Science and Technology*. 2018. Vol. 18. pp. 1-4.
10. Fraley C., Raftery A. E. How Many Clusters? Which Clustering Method? Answers Via Model-Based Cluster Analysis. *Technical Report. Department of Statistics University of Washington*. 1998. No. 329.
11. Murtagh F. A survey of recent advances in hierarchical clustering algorithms which use cluster centers. *Computer Journal*. 2020. vol. 26, no. 4, pp. 354-359.
12. Saxena A., Prasad M., Gupta A., Bharill N., Patel O. P., Tiwari A. & Lin C. T. A review of clustering techniques and developments. *Neurocomputing.* 2017. Vol. 26. p. 664-681.
13. Sneath P., Sokal R. Hierarchical Cluster Analysis: Comparison of Three Linkage Measures and Application to Psychological Data. *The Quantitative Methods for Psychology*. 2015. Vol. 11(1). pp. 8-21. DOI: 10.20982/tqmp.11.1.p008
14. Yim O., Ramdeen K. T. Hierarchical Cluster Analysis: Comparison of Three Linkage Measures and Application to Psychological Data. *The Quantitative Methods for Psychology*. 2015. Vol. 11. no. 1. pp. 8-21. DOI: 10.20982/tqmp.11.1.p008
15. Ptitsyn A., Hulver M., Cefalu W., York D., & Smith S. R. Unsupervised clustering of gene expression data points at hypoxia as possible trigger for metabolic syndrome. *BMC Genomics.* 2016. Vol. 7(1). pp. 318. doi:10.1186/1471-2164-7-318.
16. Tung A.K., Hou J., Han J. Spatial clustering in the presence of obstacles. *The 17th Intern. conf. on data engineering (ICDE'01). Heidelberg*. 2001. pp. 359–367. DOI: 10.1109/ICDM.2002.1184042.
17. Boehm C., Kailing K., Kriegel H., Kroeger P. Density connected clustering with local subspace preferences. *IEEE Computer Society. Proc. of the 4th IEEE Intern. conf. on data mining. Los Alamitos*. 2004. pp. 27–34. DOI: 10.1007/978-0-387-39940-9_605.
18. Boyko N., Kmetyk-Podubinska K., Andrusiak I. Application of Ensemble Methods of Strengthening in Search of Legal Information. *Lecture Notes on Data Engineering and Communications Technologies*. 2021. Vol. 77. pp. 188-200. URL: https://doi.org/10.1007/978-3-030-82014-5_13.
19. Boyko N., Hetman S., Kots I. Comparison of Clustering Algorithms for Revenue and Cost Analysis. *Proceedings of the 5th International Conference on Computational Linguistics and Intelligent Systems (COLINS 2021)*. Lviv, Ukraine. 2021. Vol.1. pp. 1866-1877.
20. Procopiuc C.M., Jones M., Agarwal P.K., Murali T.M. A Monte Carlo algorithm for fast projective clustering. *ACM SIGMOD Intern. conf. on management of data, Madison, Wisconsin, USA.* 2002. pp. 418–427.

АЛГОРИТМ ІЄРАРХІЧНОЇ КЛАСТЕРИЗАЦІЇ ДЛЯ ПОБУДОВИ ДЕНДРОГРАМИ ТА ПІДРАХУНКУ КЛАСТЕРІВ

Н.І. Бойко, О.А. Ткачик

Національний університет "Львівська політехніка", вул. Князя Романа, 5, Львів, 79013; Україна; nataliya.i.boyko@lpnu.ua; oleksandr.a.tkachyk@lpnu.ua

Подано вичерпний огляд ієрархічної кластеризації та побудови дендрограми з акцентом на методи визначення оптимальної кількості кластерів. Розглядаються теоретичні основи ієрархічної кластеризації та процес побудови дендрограм, а також описуються кілька популярних методів визначення кількості кластерів. Метою дослідження є застосування різних алгоритмів для побудови дендрограм та визначення оптимальної кількості кластерів в ієрархічній кластеризації. Дослідження має на меті забезпечити теоретичні основи ієрархічної кластеризації, пояснити процес побудови дендрограми та обговорити різні методи визначення кількості кластерів. Об'єкт дослідження включає як ієрархічні, так і неієрархічні алгоритми кластеризації, і має на меті забезпечити вичерпний і практичний посібник із різних методів кластеризації для дослідників і практиків у цій галузі. Робота зосереджена як на методах роздільної, так і на агломераційній кластеризації, а також на дендрограмі, перевагах і недоліках кожного методу, а також на тому, як дендрограми використовуються для візуалізації результатів ієрархічної кластеризації. Він також забезпечує порівняння ієрархічної кластеризації з неієрархічною кластеризацією, зокрема алгоритмом K-середніх, і обговорює їхні відповідні переваги та недоліки. Однією з ключових переваг ієрархічної кластеризації є те, що вона не вимагає від користувача заздалегідь вказувати кількість кластерів, як у випадку з неієрархічною кластеризацією. Зазначається, що для визначення відповідної кількості кластерів можна використовувати дендрограму. В результаті відзначається корисність ієрархічної кластеризації для ряду додатків, зокрема для пошукового аналізу даних. У дослідженні також розглядаються основні методи визначення найбільш схожих об'єктів і кластерів. Крім того, у статті надається огляд методу кластеризації K-середніх і порівнюється його з ієрархічною кластеризацією.
Ключові слова: агломеративна кластеризація, роздільна кластеризація, ієрархічна кластеризація, k-середні.