Software for Processing the Results of Psychological Research*

Viktoriya V. Buchatskaya 1[0000-0001-6945-6901], Pavel Yu. Buchatskiy 1[0000-0002-3161-6567],
Ekaterina V. Makrischeva 1[0000-0002-2935-2339]

[1]Adyghe State University, Maykop, Russia
buch_vic@mail.ru
butch_p99@mail.ru
makrishcheva97@mail.ru

Abstract. When processing the results of research in psychology, it becomes necessary to divide the set of respondents into groups according to several criteria. Cluster analysis techniques are used to perform this procedure. Cluster analysis algorithms today have many different implementations. Features of psychological studies do not allow to fully automate the processing of results. In this regard, it is relevant to develop a software application for cluster analysis and visualization of results, which does not require special mathematical knowledge in the process of using and has a convenient and understandable user interface. The article describes the structure and results of a software application that implements cluster analysis of the results of psychological research. The application implements the k-means algorithm, DBSCAN algorithm, agglomerative and spectral clustering, assessed the quality of clustering by the indicated algorithms, and the best solution is selected. Results visualization is provided for the analysis of the split. The software application is implemented in Python using the libraries NumPy, Pandas, Matplotlib, SciPy, Scikit-learn (Sclearn), Tkinter.

Keywords: Cluster Analysis, K-Means Algorithm, DBSCAN Algorithm, Algorithmic and Spectral Clustering.

1 Introduction

Quite often, in the course of processing the results of research in psychology, it becomes necessary to divide the set of respondents into groups according to several criteria. To perform this procedure, cluster analysis methods are used. Cluster analysis algorithms today have many different implementations. However, the peculiarities of psychological research itself do not allow fully automate the process of processing the results. In this regard, it is relevant to develop a software application for cluster analysis and visualization of results, which does not require special mathematical knowledge in the process of using and has a convenient and understandable user interface.

The research task is set as follows: develop a software application for the implementation of cluster analysis algorithms, perform clustering, compare the results and determine the most appropriate cluster analysis algorithm for the results of psychological research.

2 Theoretical part

The formal statement of the clustering problem has the form:

Let *X* be a set of objects, *Y* a set of numbers (names, labels) of clusters. The function of the distance between objects $\rho(x, x')$ is given. There is a finite training set of objects $X^m = \{x_1,\dots, x_m\} \subset X$. It is required to split the sample of clusters so that each cluster consists of objects close in the metric ρ, and objects of different clusters differ significantly. Moreover, each object $x_i \in X^m$ is assigned a cluster number y_i.

The process of implementing cluster analysis can be represented in the following stages [1, 2]:

- selection of an initial set of objects;
- determination of the set of variables by which the objects in the sample will be evaluated; if necessary, normalization of variable values;
- calculating the values of the measure of similarity between objects;
- the application of the method of cluster analysis to create clusters;
- presentation of analysis results.

The clustering algorithm is a function a: $X \to Y$, which assigns to any object $x \in X$ the cluster number $y \in Y$. In some cases, the set *Y* is known in advance, but more often the task is to determine the optimal number of clusters from the point of view of one or another criterion of clustering quality.

The advantages and disadvantages of the most commonly used clustering algorithms are presented in Table 1.

Table 1. Advantages and disadvantages of clustering algorithms.

Algorithm	Advantages	Disadvantages
Agglome-rative	- builds an optimal partition; - getting all partitions in the form of a dendrogram.	- it is necessary to set the threshold k; - computational convergence; - clusters do not overlap; - computational complexity.
Mean Shift	-is an application-independent tool suitable for analyzing real-world data; - does not take a predefined form in data clusters; - simple implementation; - resistant to the choice of initial data.	- computational complexity; - you need to choose the initial parameters.
CURE	- high-level clustering even in the presence of outliers; - selection of clusters of complex shapes and different sizes.	- works only with numerical data; - the need to set threshold values and the number of clusters.

BIRCH	- two-stage clustering: - clustering of large amounts of data; - works on a limited amount of memory: - is a local algorithm: - can work with one scan of the input dataset: - data can be unevenly distributed in space; - treats areas of high density as a single cluster.	- works only with numerical data; - well distinguishes only clusters of a convex or spherical shape; - the need to set threshold values.
MST	- works with large sets of arbitrary data; - selects clusters of arbitrary shape; - selects the best from several optimal solutions.	- sensitive to emissions.
k-means	- ease of use; - speed of work; - clarity and transparency of the algorithm.	- sensitive to emissions; - slow work at large volumes; - it is necessary to set the number of clusters; - impossibility of application on data where there are intersecting clusters; - achievement of a global minimum is not guaranteed; - the operation of the algorithm strongly depends on the chosen initial cluster centers, the optimal value of which cannot be known in advance.
PAM	- ease of use; - speed of work; - clarity and transparency of the algorithm; - less sensitive to outliers compared to k-means	- it is necessary to set the number of clusters; - slow work on large databases.
CLOPE	- clustering of huge sets of categorical data; - scalability; - speed of work; - the quality of clustering, which is achieved by using the global optimization criterion based on maximizing the gradient of the height of the cluster histogram; - easy to calculate and interpret; - small amount of resources; - automatically selects the number of clusters; - regulated by one parameter - the repulsion coefficient.	- uncertainty with outliers.
DBSCAN	- does not require specification of the number of clusters in the data a priori, unlike the k-means method;	- not completely unambiguous - edge points that can be reached from more than one cluster may belong to any of

	- can find clusters of arbitrary shape; - has the concept of noise and is resistant to emissions; - requires only two parameters and is mostly insensitive to the order of points in the database; - designed for use with databases that allow you to speed up queries in a range of values, for example, using an R * -tree; - the parameters can be set by experts in the field in question if the data is well understood.	these clusters, depending on the order in which the points are viewed; - quality depends on distance measurement; - cannot cluster datasets well with large differences in density.
Spectral	- ease of use; - speed of work; - clarity and transparency of the algorithm.	- works only with numerical data; - it is necessary to set the number of clusters.

From the information given in the table, we can conclude that there is no single universal clustering algorithm. When using any algorithm, it is necessary to evaluate its advantages and disadvantages in solving a specific problem.

3 Practical part

Based on the analysis of the information presented in Table 1, to solve the problem, we will use the following algorithms: generalized agglomerative clustering algorithm, k-means algorithm, DBSCAN, and spectral algorithm. They are the most common, allow you to work correctly with a large amount of data, and are representatives of different groups of clustering algorithms. [3, 4, 5] The input data is a two-dimensional array of *n* records, containing user data of the type of natural numbers.

The developed software application consists of 5 blocks: formation of initial data; data preprocessing; clustering; assessment of the quality of clustering; visualization. The structure of the software application is shown in Fig. 1.

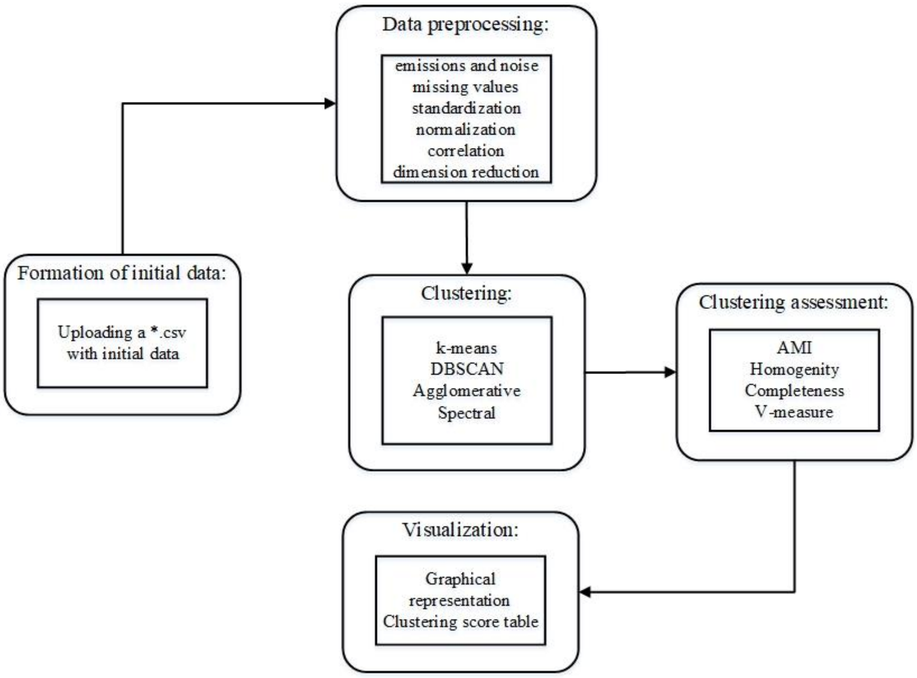


Fig. 1. Structure of the software application.

The initial data is loaded into the application as a file with the *.csv extension. Before the data is subjected to cluster analysis, it must be brought to a form following the requirements determined by the specifics of the task at hand, that is, to perform a preliminary processing procedure, which includes the following stages[6, 7, 8]:

- check for missing data;
- cleaning from outliers and noise in data;
- correlation;
- standardization;
- normalization;
- reduction of dimension.

The clustering block implements 4 clustering algorithms: k-means, DBSCAN, Agglomerative and spectral clustering. The listed algorithms process a given set of data. Clustering results are assessed based on the following metrics: AMI, homogeneity, completeness, v-measure[8].

The clustering result is visualized as a scatterplot, and the clustering score is visualized as a table.

To solve the problem, the high-level programming language Python was chosen. During the development of the software application, the interactive Jupyter Notebook shell was used, which contains the command line and the text editor IDE Python 3.8.

Three buttons were used for the software application: File download; Preliminary processing; Start.

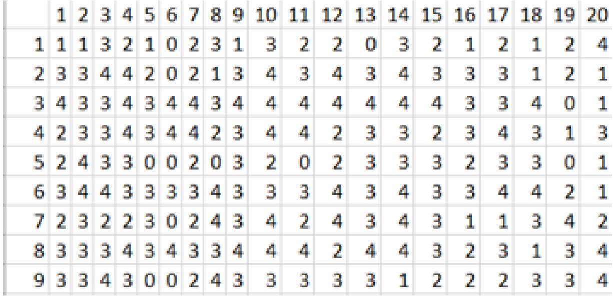
The ComboBox widget was used to select the clustering method, and the Check Button was used to evaluate and visualize. Clustering results, i.e. visualization of the results are displayed in an additional window.

By clicking on the Load File button, the insertText() function is called. The pd.read_csv() function implements a call to the loading dialog box as a file with the *.csv extension. The input data is a two-dimensional array of n records of the type of natural numbers. After loading the initial data, the application form displays the full path to the file containing the data.

By clicking on the "Data preprocessing" button, the pre_treatment() function is called. It detects outliers and noise, missing values, standardizes and normalizes data, and reduces dimensionality. The preprocessing results are not displayed on the screen, they are entered into the data file.

Next, the m_clust() function is triggered, which implements the choice of the clustering method, the clustering assessment, and the visualization of the result. The selection of the indicated positions is carried out from the drop-down lists. Only one clustering method can be selected in one pass of the program. At the same time, the user has the opportunity to choose a quality assessment method and a method for visualizing the results. After completing all the necessary parameter settings in the window, by clicking on the "Start" button, the m_clust() function is called. The result of its work is the division of a given set of objects into clusters, visualization of the clustering results in the form of a scatter diagram, as well as the output of the clustering assessment results in the form of a table in an additional window. The splitting results are also written to a new * .csv file, while the data is distributed across clusters with an indication of the number of objects that fall into one or another cluster.

To demonstrate the operation of the application, as input data, we used the results of the research of the candidate of psychological sciences, associate professor N.V. Kovaleva (Maikop, ASU) on the topic «Subjective identity of a person in an innovative environment». The initial data included personal information about each respondent and five blocks of questions: necessity, motives, impact, and emotional-activity response to uncertainty. To form the original dataset, personal information was removed from them. As a result, the input data took the form shown in Fig. 2.



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	1	1	3	2	1	0	2	3	1	3	2	2	0	3	2	1	2	1	2	4
2	3	3	4	4	2	0	2	1	3	4	3	4	3	4	3	3	3	1	2	1
3	4	3	3	4	3	4	4	3	4	4	4	4	4	4	4	3	3	4	0	1
4	2	3	3	4	3	4	4	2	3	4	4	2	3	3	2	3	4	3	1	3
5	2	4	3	3	0	0	2	0	3	2	0	2	3	3	3	2	3	3	0	1
6	3	4	4	3	3	3	3	4	3	3	3	4	3	4	3	3	4	4	2	1
7	2	3	2	2	3	0	2	4	3	4	2	4	3	4	3	1	1	3	4	2
8	3	3	3	4	3	4	3	3	4	4	4	2	4	4	3	2	3	1	3	4
9	3	3	4	3	0	0	2	4	3	3	3	3	3	1	2	2	2	3	3	4

Fig. 2. Input data.

The source data table consisted of 149 rows (respondents) and 20 columns, i.e. the number of test questions, the first column is responsible for the number of the respondent, and the first line is for the number of the question.

On the graphs displaying the clustering results shown in Fig. 3, 4 it can be seen that the input data was divided into five clusters.

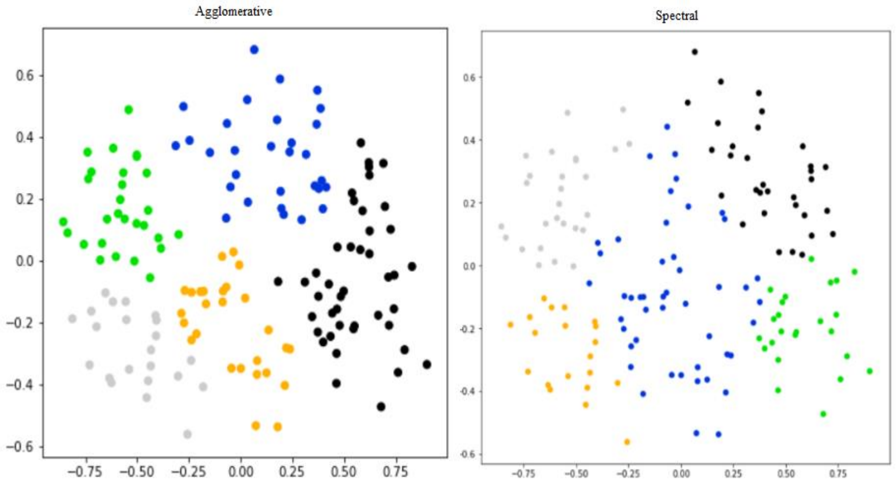


Fig. 3. Partitioning by the method of agglomerative and spectral clustering.

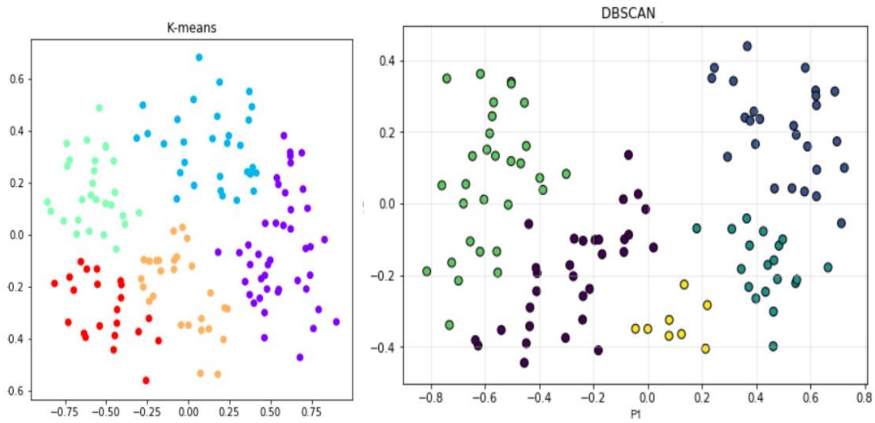


Fig. 4. K-Means partitioning and DBSCAN.

By the arrangement of the elements in the clusters relative to the center, we can say that some points of the clusters are located at a distance, this indicates that these elements are at the maximum distance from their cluster.

The values of the clustering quality metrics are shown in Table 2.

Table 2. Clustering estimates.

Algorithm	AMI	Homogenity	Completeness	V-measure
k-means	54,1 %	30,7%	100 %	47,4 %
DBSCAN	68,5 %	34,3%	100 %	50,8 %
Agglomerative	74,2 %	63,6 %	100 %	14,7 %
Spectral	31,8 %	51,0 %	100 %	33,2 %

4 Conclusion

From the information presented, it can be seen that the indicator for the first assessment metric is higher for the agglomerative clustering method is 74.2%. It says that the elements of one cluster differ from the elements of another cluster by the specified amount. According to the second metric, agglomerative clustering has the highest indicator, and it is 63.6%. This metric says that the points of the cluster must belong to this particular cluster. As for the third metric, the completeness of clustering, for all methods, is 100%, which means that, as accurately as possible, he estimated the belonging of a cluster point to this cluster. According to the value of the fourth metric, which determines the percentage of similarity of clusters in each other, the agglomerative method of clustering also gives the best result of 14.7%. Thus, on a given set of initial data, the best clustering results were shown by the agglomerative clustering method.

The software application is implemented in Python using the NumPy, Pandas, Matplotlib, SciPy, Scikit-learn (Sclearn), Tkinter modules. [9, 10]

References

1. Lab, K.: CLUTO - Software for Clustering High-Dimensional Datasets. URL: http://glaros.dtc.umn.edu/gkhome/cluto/cluto/overview.
2. Mirkin, B.: Mathematical Classification and Clustering. B. Mirkin. Springer US, 448 p. (1996)
3. Paklin, N.: Clustering Algorithms on the Data Mining Service. URL: https://basegroup.ru/community/articles/datamining
4. Buchatskaya, V.V., Buchatsky, P.Yu., Gushchin, K.A.: A software application for data clustering. System Administrator. Moscow, 2017, vol. 1-2, pp. 170-171. (2017)
5. Chris, A.: Machine Learning with Python Cookbook: Practical Solutions from Preprocessing to Deep Learning, O'Reilly Media, N.Y., 2018, pp. 366. (2018)
6. Fasulo, D.: An Analysis of Recent Work on Clustering Algorithms. URL: https://www.researchgate.net/publication/2279437_An_Analysis_of_Recent_Work_on_Clustering.
7. Wu, J.: Advances in K-means Clustering. Springer-Verlag Berlin Heidelberg, 180 p. (2012)
8. Zakharov, K.: Application of k-means clustering in psychological studies. The Quantitative Methods for Psychology, vol. 12 (2), 2016, pp. 87–100. (2016)
9. Charu, C.Aggarwal, Chandan, K. Reddy.: Data Clustering. Algorithms and Applications. Chapman and Hall/CRC, 2013, pp. 652. (2013)
10. Matthes, E.: Python Crash Course, 2nd Edition: A Hands-On, Project-Based Introduction to Programming. No Starch Press; 2nd Edition, 2019, pp. 544. (2019)