УДК 004.67
DOI https://doi.org/10.24144/2616-7700.2023.42(1).129-147

**N. I. Boyko[1], O. A. Tkachyk[2]**

[1] Lviv Polytechnic National University,
Associated Professor at the Department of Artificial Intelligence,
Ph.D.
nataliya.i.boyko@lpnu.ua
ORCID: https://orcid.org/0000-0002-6962-9363

[2] Lviv Polytechnic National University,
Graduate student at the Department of Artificial Intelligence,
oleksandr.a.tkachyk@lpnu.ua
ORCID: https://orcid.org/0000-0002-0728-4208

# CLUSTERING ALGORITHMS AND METHODS FOR DIVERSE DATA

The study is dedicated to the comprehensive investigation of clustering methods for diverse data. The research is focused on the problems of graphic format algorithms, which is conditioned by the presence of 12 different features for clustering, 7 of which were categorical. The data is presented along 12 axes in a graphical format. To solve the problem the PCA algorithm was applied with further transformation of categorical features into numerical for dimensionality reduction to 2 components and further orthogonal superimposition of clusters on them. Clustering using the k-prototype method was provided. A sixfold decrease in PCA algorithm has drawbacks such as enormous data loss which was presented. Based on the list of conducted experiments on hierarchical clustering the pros and cons can be seen for this approach. The complexity of clustering which consists in representation of results from the analysis of big data was provided. The KAMILA algorithm that is based on distributed computing models MapReduce and gives a significant advantage was described.

**Keywords:** expectation-maximization, Structural equation modeling, KAy-means for MIxedLArge data, Lowest common ancestor, self-organizing map, Adaptive resonance theory, Kernel Density Estimation.

**1. Introduction.** Clustering is a technique used in data analysis and machine learning to divide the given data into groups of similar data points based on their characteristics or attributes which are called clusters. The goal of clustering is to identify patterns or structures in the data that can help to better understand the underlying data distribution, to make predictions or to enable efficient processing. Algorithms can be significantly different in that sense that what has to be included to each cluster and how to find their dependencies more effectively. Among the list of popular conceptions of clusters there are groups with the elements which can be built based on distance between them, density of areas in data spaces, intervals or specific statistical distribution [1, 4].

**The purpose of study** — applying different algorithms and methods for clustering diverse data.

**The object of this study** is to conduct an analysis and compare applied methods or algorithms, therefore to identify the most effective one that showed the most precise results.

Since the main goal in clustering analysis is to find a certain number of groups of objects which are similar between each other inside the specific group and which

are most different compared to other groups, data processing is required to identify the main direction of how it will be processed later. In order to find dependencies in objects with diverse data and apply clustering feature engineering should be applied.

Most of the algorithms used for data clustering are based on data adaptation for clustering methods for repetitive data. There are two main approaches by which clustering is performed:

1) Transforming numerical data into categorical features with further applying clustering methods for that feature;
2) Transforming categorical features into numeric forms, scaling them and applying clustering methods for numeric data.

In current investigation next research methods will be performed:

- Comparison — indicating the differences between different clustering methods.
- Calculations — searching for numerical features to use in clustering methods such as job execution time etc.
- Analysis — decomposing clustering methods into several properties or characteristics.
- Formalization — visualizing clustering algorithms as mathematical formulas.

From the practical point of view the current research can be used for diverse data clustering in different scopes. By getting familiar with this paper researchers will be able to choose optimal clustering methods based on its properties.

**2. Analysis of recent resources.** For better understanding of the principles and aspects of diverse data clustering and summarizing existing approaches, algorithms and methods which are used for data clustering, the analysis of recent resources was done. The results of analysis will be used during investigations described in next sections.

The paper [2, 11] has described different developed models and algorithms of clustering analysis, described problems of some algorithms and proposed possible solutions to solve them. As the conclusion from described content it can be said that applying ensemble methods is very promising and effective for processing diverse data.

In the research [5, 12] the main approaches of diverse data clustering is reviewed, the abilities of usage of similar algorithms is described. Application of different distances between diverse data and possible hierarchical distribution is reviewed. The work describes leading tasks and open questions in diverse data clustering areas.

Researchers [3, 15] investigated the approach of Dharmendra S. Modhia and Scott Spangler for clustering diverse data in their article. They described KAMILA algorithm and its theoretical implementation using R language with further algorithm simulation using random data. The results of clustering on "balanced" and "unbalanced" data sets were analyzed. The possible implementation of the algorithm for analyzing big data using Apache Hadoop and its results were described.

Research [6] is interesting by its analysis of basic clustering algorithms such as k-means, density-based clustering and agglomerative clustering with further implementation using C language.

In the article [7, 13] the main diverse data clustering problems were described. Applying hybrid distance techniques were analyzed. Basic steps such as data transformation with discretization and productivity of k-modes and LCA algorithms were described. As a solution of the described problems KAMILA algorithm was applied and analyzed.

Authors [8, 10] proposed a model based on mixed data clustering using SEM algorithm. Advantages and possibilities of work with incomplete data of provided solutions were described.

**3. Methods and tools of research.** The approach to clustering different types of data involves processing the data and using clustering algorithms. Clustering algorithms can be divided into several categories, including partitioning, hierarchical, model-based, neural network-based, and others.

**3.1. Data processing.** Clustering datasets in a given dataset is almost always dependent on the structure and types of data that are present within it. If there is no common structure or if the features are not well defined, clustering is likely to result in inaccurate results, as the boundaries between clusters are difficult to determine. Therefore, to achieve better clustering results, it is necessary to properly process our datasets. Almost always, it is necessary to find a balance between information loss and distortion. Data preprocessing is a critical step in clustering, as it can help to improve the quality of the clusters and reduce the impact of noise and outliers in the data. Data preprocessing techniques can include normalization, feature scaling, dimensionality reduction, and handling of missing values. Data can be categorized as categorical or numerical. Numerical data consists of ordinary numbers that represent a particular feature, such as the quantity of something, a certain distance, age, etc. Categorical data, on the other hand, represents particular groups or categories, such as race, gender, or blood groups. To find common characteristics among numerical data, algorithms are typically used to calculate distances between them. However, it is more difficult to do so with categorical data, as there are no numerical values to calculate distances [9, 10].

In all cases of clustering different types of data, the dataset contains both categorical and numerical data. This is the main challenge in such clustering scenarios. When considering the simplest approaches for finding common characteristics among different types of data, the following method is commonly used: the numerical and categorical data are separated and the Euclidean distance is calculated between the numerical data, while the Hamming distance is calculated between the categorical data. The next step is mixing, where the metrics obtained from the distance calculations of the two types are combined to find a distance between the different types of data. This is done by combining the metrics obtained from the calculations of the two types, such as the Euclidean and Hamming distances, to find a single distance metric that captures the similarities and differences between the different types of data. Of course, direct mixing will not help because the result would be no precise [11]. The reason why it is better to choose different distance metrics for numerical and categorical data is because they have different characteristics and require different approaches for calculating distances. For example, numerical data is continuous and can be measured on a scale, while categorical data consists of non-numerical values that represent particular categories or groups.

Therefore, finding a common metric that works well for both numerical and categorical data is not always straightforward and can be a challenging task.

**3.2. Divisive clustering algorithms.** The most well-studied methods for clustering different types of data belong to the family of algorithms that partition data into clearly defined groups using either purely numerical data (k-means) or purely categorical data (k-modes). The main idea behind such algorithms is to

determine the center of a cluster using numerical or categorical data, compute distances from the centroid to the objects being studied, and then process the mixed data types to find the local minimum.

The main advantages of such algorithms are their speed and ability to be parallelized (e.g., using MapReduce), which makes them suitable for working with large datasets. Some well-known algorithms in this family include Huang's, Ahmad and Dey's, Zhao, Modh and Spangler's, and Ren's [5] algorithms.

Z. Huang's algorithm [6, 11], also known as the k-prototype method, uses the Euclidean distance from the mean values to numeric data and the Hamming distance for the most common data to categorical data. However, the resulting cluster centers may not accurately reflect the clusters due to potential loss of information caused by the Hamming distance, which only considers the presence or absence of agreement between two categorical values.

Sartaj Ahmad and Gurav Deh's approach [7, 15] involves developing a new cost function and distance calculation to address the limitations of the k-prototypes algorithm. They calculate similarity between categorical data based on co-occurrence of values with other features, and also weight numerical features using this method. Overall, this algorithm performs better than the k-prototypes algorithm.

Another approach is the algorithm by V. Zhao [4], which, to avoid the drawbacks of the Hamming distance, used the frequency of occurrence of categorical data to determine the cluster centers, which also resulted in better results than using the k-prototypes algorithm.

Dharmendra S. Modha and Scott Spangler proposed an algorithm in which each data point lies in different spaces. The calculation of weights is based on the measure of distortion between different spaces of objects. For numerical features, the squared Euclidean distance is used, and for categorical features, the cosine distance is used.

M. Ren [6] uses the approach of building cluster centers based on the k-prototypes algorithm, further developing the idea of Sartaj Ahmad and Gurav Dei [7, 14] by applying a Gaussian filter to the overall distance values and combining the determination of the cluster center with the feature weights to create a new cost function. Since the initial weights are initialized with random values, this algorithm may produce different results after several runs with the same input data.

The K-prototypes algorithm defines G virtual individuals (or prototypes) as the centers of groups, constructed from the mean values per group for numerical variables and the mode per group for categorical variables. The distance between two subjects X and Y is determined as follows (Formula 1):

$$d_2(X,Y) = \sum_{j=1}^{q} (x_j - y_j)^2 + \gamma \sum_{j=q+1}^{p} \delta(x_j, y_j) \tag{1}$$

where $\sum_{j=1}^{q} (x_j - y_j)^2$ is the squared Euclidean distance for continuous variables; $\gamma \sum_{j=q+1}^{p} \delta(x_j, y_j)$ — Hamming distance.

The weight $\gamma$ is used to avoid bias towards any type of attribute. It can be specified by the user or estimated using the combined variance of the data.

The minimization criteria is the total sum of distances (TSD) between the sub-

jects and the prototype of the class $b_g$ to which they belong (Formula 2):

$$TSD = \sum_{g=1}^{G} \sum_{x \in C_g} \left( \sum_{j=1}^{q} (x_j - b_{g,j})^2 + \gamma \sum_{j=q+1}^{p} \delta(x_j, b_{g,j}) \right) \tag{2}$$

The K-prototypes algorithm is similar to k-means in practice: initial prototypes G are chosen as temporary cluster centers, then each subject is assigned to the nearest prototype. When all subjects are assigned, prototypes are updated to reflect their optimal class. Then subjects are reassigned to the updated prototypes if necessary, and the process repeats until the distribution becomes stable.

**3.3. Hierarchical clustering.** Hierarchical clustering methods create a hierarchy of clusters organized from top to bottom (or bottom to top). To create clusters, hierarchical algorithms require the following [13]:

1. Similarity matrix — built by finding similarities between each pair of data points. The choice of similarity metric (for building the similarity matrix) affects the shape of the clusters;
2. The linkage criterion — it determines the distance between sets of observations as a function of the pairwise distance between observations.

Most hierarchical clustering algorithms have a high computational complexity of $O(n^3)$ and require $O(n^2)$ memory, where $n$ is the number of data points. For constructing a similarity matrix in the case of heterogeneous data types, the Gower distance can be used.

The Gower distance can be used to measure the dissimilarity between two records that may contain a combination of logical, categorical, numerical, or text data. The distance is always a number between 0 (identical data) and 1 (maximally different data). For numerical data, the normalized Manhattan distance is used, for ordinal data the variable is first ranked and then the Manhattan distance is used. For nominal data, the $k$ categories are first transformed into $k$ binary columns, and then the Dice coefficient is used.

Strategies for hierarchical clustering can be divided into two types

1. Agglomerative (or bottom-up) clustering, where each point starts as its own cluster and pairs of clusters are merged as the algorithm moves up the hierarchy.
2. Divisive (or top-down) clustering, where all points start in a single cluster and recursive splitting occurs as the algorithm moves down the hierarchy.

Agglomerative clustering starts with $N$ clusters (one for each subject), and at each step, the two closest clusters are merged until only one cluster remains. The sequence of mergers is represented on a dendrogram to facilitate the choice of an optimal number of clusters. In general, the best cluster allocation is the one that precedes the first significant increase in within-cluster variance [1, 3].

Let's suppose that at a certain stage of aggregation, clusters $C_i$ and $C_j$ are the next ones to be merged. To determine the distance of the merged cluster $C_i \cup C_j$ to any other cluster $C_k$, the similarity matrix needs to be updated using a single linkage method, which belongs to the family of Lance-William's algorithms (Formula 3):

$$d\left(C_i \cup C_j, C_k\right) = \alpha d\left(C_i, C_k\right) + \beta d\left(C_j, C_k\right) - \eta d\left(C_i, C_j\right). \tag{3}$$

The coefficients $\alpha$, $\beta$ and $\eta$ depend on the aggregation method used. These methods for calculating distances between clusters are called linkage criteria. Using the Ward aggregation method, Formula 3 takes the following form (Formula 4):

$$d(C_i \cup C_j, C_k) = \frac{n_i + n_k}{n_i + n_j + n_k} d(C_i, C_k) + \\ + \frac{n_j + n_k}{n_i + n_j + n_k} d(C_j, C_k) - \frac{n_k}{n_i + n_j + n_k} d(C_i, C_j). \tag{4}$$

where $n_i$, $n_j$ and $n_k$ represents corresponding sample sizes $C_i$, $C_j$ and $C_k$.

**3.4. Clustering algorithms based on models.** Clustering algorithms based on models assume that a data point corresponds to a model, which is typically a statistical distribution. The models are usually chosen by the user, which can lead to undesirable results if the model parameters are not well-suited to the data. Algorithms in this family are generally slower than distribution-based algorithms but can avoid undesirable information loss. Some well-known algorithms in this family include Autoclass, the Everett algorithm, ClustMD, KAMILA, the Mustaki and Papageorgiou model, Brown and Mac-Nicholas model, and the Rajan and Bhattarcharyya algorithm.

The Autoclass algorithm [4] performs clustering by integrating a mixture model and Bayesian methods with a prior distribution for each feature.

B. S. Everitt [2] proposed a model-based clustering algorithm for heterogeneous data using threshold values for categorical data. Due to its high computational cost, this method is only useful for datasets that contain few categorical features.

I. Mustaki and I. Papageorgiou [5] used LCM for heterogeneous data by transforming categorical features into numerical values from 1 to $q$. Polynomial distributions were used for categorical features, and normal distributions for numerical features. Similar solutions were proposed by R. P. Brown and P. D. McNicholas [6], who developed a model that combines hidden features and uses the EM algorithm for model fitting.

The ClustMD algorithm [8] uses a hidden variable model for clustering heterogeneous data. It assumes that the hidden variable, along with a Gaussian mixture distribution, represents a particular data point. The EM algorithm is also used to estimate the parameters of the model. For categorical data, the Monte Carlo EM algorithm is used. The main problem with this method is the increased computational cost as the number of features increases.

V. Rajan and S. Bhattacharya [9] introduced an algorithm based on a mixture of Gaussian copulas, which can model dependencies between both categorical and numerical features. This method is faster on a significant number of datasets.

A. Foss, M. Markatou, B. Ray, and A. Heching [10] developed the KAMILA algorithm for clustering heterogeneous data. This method combines two different clustering algorithms, namely the k-means algorithm and the Gaussian mixture multinomial model. Like the k-means algorithm, KAMILA does not make significant parametric assumptions for numerical features, but instead uses the properties of Gaussian mixture multinomial models to balance the effects of numerical and categorical data without weighting them.

**3.5. Clustering algorithms based on neural networks.** Most research on clustering of heterogeneous data using neural networks focuses on the use of self-organizing maps (SOM) and adaptive resonance theory (ART). However, the use of SOM can lead to unpredictable results, such as poor data representation and

failure to match the data distribution structure. On the other hand, ART is more computationally complex but can be quite effective. ART neural network models work on the principle that provides adequate ways of clustering data alongside other popular methods that help to reduce the dimensionality of data sets. Compared to k-means clustering, ART is a parameterized algorithm. In k-means, the number of clusters has to be specified prior to the calculations, unlike ART which has a certain threshold. With this threshold, clusters can be created in real-time. Additionally, it can determine how loose or tight a cluster will be. One of the main drawbacks of using a single threshold is that it is applied to all possible clusters, for example, a threshold value may not lead to high accuracy for both dense and sparse clusters. To address this, different thresholds can be used for each cluster. To determine this threshold, the Particle Swarm Optimization (PSO) machine learning method is used. PSO performs a search for global maximum or minimum. The position of particles on each iteration is evaluated according to the fitness function. If the swarm finds the best particle, a new threshold coefficient is calculated.

When ART creates a new cluster, the threshold coefficient increases and a new swarm is initialized to optimize the performance of ART. This operation helps to expand the threshold coefficients for each cluster, optimizing each threshold for its respective cluster.

There are also other methods of traditional neural network clustering that can be applied to clustering heterogeneous data, such as Adaptive Subspace SOM, ARTMAP, and Vector Quantization.

**4. Experiments.** For applying clustering algorithms, a dataset of cardio indicators of patients in a hospital was chosen. The dataset contains 5 numeric and 7 categorical features (Fig. 1). Next, the dataset needs to be processed, for which we will remove records with missing values and convert categorical features to factors for processing categorical data in the *R* programming language.

| | age | gender | height | weight | ap_hi | ap_lo | cholesterol | gluc | smoke | alco | active | cardio |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 18393 | men | 168 | 62 | 110 | 80 | normal | normal | smokes | doesn't drink alco | 1 | 0 |
| 2 | 20228 | women | 156 | 85 | 140 | 90 | well above normal | normal | smokes | doesn't drink alco | 1 | 1 |
| 3 | 18857 | women | 165 | 64 | 130 | 70 | well above normal | normal | smokes | doesn't drink alco | 0 | 1 |
| 4 | 17623 | men | 169 | 82 | 150 | 100 | normal | normal | smokes | doesn't drink alco | 1 | 1 |
| 5 | 17474 | women | 156 | 56 | 100 | 60 | normal | normal | smokes | doesn't drink alco | 0 | 0 |
| 6 | 21914 | women | 151 | 67 | 120 | 80 | above normal | above normal | smokes | doesn't drink alco | 0 | 0 |
| 7 | 22113 | women | 157 | 93 | 130 | 80 | well above normal | normal | smokes | doesn't drink alco | 1 | 0 |
| 8 | 22584 | men | 178 | 95 | 130 | 90 | well above normal | well above normal | smokes | doesn't drink alco | 1 | 1 |
| 9 | 17668 | women | 158 | 71 | 110 | 70 | normal | normal | smokes | doesn't drink alco | 1 | 0 |
| 10 | 19834 | women | 164 | 68 | 110 | 60 | normal | normal | smokes | doesn't drink alco | 0 | 0 |
| 11 | 22530 | women | 169 | 80 | 120 | 80 | normal | normal | smokes | doesn't drink alco | 1 | 0 |
| 12 | 18815 | men | 173 | 60 | 120 | 80 | normal | normal | smokes | doesn't drink alco | 1 | 0 |
| 13 | 14791 | men | 165 | 60 | 120 | 80 | normal | normal | smokes | doesn't drink alco | 0 | 0 |
| 14 | 19809 | women | 158 | 78 | 110 | 70 | normal | normal | smokes | doesn't drink alco | 1 | 0 |
| 15 | 14532 | men | 181 | 95 | 130 | 90 | normal | normal | doesn't smoke | drinks alco | 1 | 0 |
| 16 | 16782 | men | 172 | 112 | 120 | 80 | normal | normal | smokes | doesn't drink alco | 0 | 1 |
| 17 | 21296 | women | 170 | 75 | 130 | 70 | normal | normal | smokes | doesn't drink alco | 0 | 0 |
| 18 | 16747 | women | 158 | 52 | 110 | 70 | normal | well above normal | smokes | doesn't drink alco | 1 | 0 |
| 19 | 17482 | women | 154 | 68 | 100 | 70 | normal | normal | smokes | doesn't drink alco | 0 | 0 |
| 20 | 21755 | men | 162 | 56 | 120 | 70 | normal | normal | doesn't smoke | doesn't drink alco | 1 | 0 |

Figure 1. Dataset with patient characteristics

So, let's assume that our dataset consists of $N$ independent and identically distributed observations of a n-dimensional random vector of variables. To cluster this dataset, we will apply the KAMILA algorithm, which estimates unknown pa-

rameters using an iterative process similar to the EM algorithm. In other words, KAMILA also performs iterative estimation, where each iteration consists of two steps: partitioning and estimation. The partitioning step assigns each observation to a certain cluster, while the estimation step re-estimates certain cluster parameters including the newly added observation.

For example, let's take one iteration of the algorithm. First, we need to calculate the Euclidean distance for each numerical feature to use the Gaussian kernel for estimating minimum distances using formula 5:

$$f_R^{(t)}(r) = \frac{1}{Nh^{(t)}} \sum_{\ell=1}^{N} k\left(\frac{r - r_\ell^{(t)}}{h^{(t)}}\right). \tag{5}$$

We assume that the categorical features are independent of the numerical ones, and the algorithm estimates them based on the numerical ones. The algorithm makes initial assumptions, so on some iterations, the data may differ significantly and be difficult to process (Figure 2).

```
Categorical variable 1: num [1:4, 1:2] 0.356 0.357 0.363 0.357 0.644 ...
Categorical variable 2: num [1:4, 1:2] 0.356 0.357 0.363 0.357 0.644 ...
Categorical variable 3: num [1:4, 1:3] 0.0846 0.0823 0.0879 0.083 0.8282 ...
Categorical variable 4: num [1:4, 1:2] 0.108 0.109 0.111 0.11 0.892 ...
Categorical variable 5: num [1:4, 1:2] 0.9222 0.9247 0.919 0.9242 0.0778 ...
Categorical variable 6: num [1:4, 1:2] 0.217 0.21 0.208 0.212 0.783 ...
Categorical variable 7: num [1:4, 1:2] 0.497 0.497 0.495 0.501 0.503 ...
```

Figure 2. Results of computing weights for categorical features.

In the given dataset, there are 7 categorical features. As the algorithm evaluates each observation based on its numerical values with projection onto categorical features, it calculates weights for each categorical feature. Then, it combines the information and assigns the observation to the cluster that was selected based on the collected information, and re-estimates the clusters.

The KAMILA.r package uses a straight stopping rule for clustering if certain groups of observations remain unchanged from one iteration to the next.

It can also be noted that our dataset contains 70,000 records, so the stopping rule will be quite useful in this case.

To visualize the clusters based on categorical and numerical data, a two-dimensional scatter plot with components on the x and y axes was used. These two components are the result of applying principal component analysis (PCA) to the data. They can be characterized as linear combinations of the input variables that account for most of the variability in the observations. From Figure 3, we can see that the data is separated into 3 clusters, and for analysis we will apply the KAMILA algorithm and the k-prototypes algorithm with manually specified number of clusters and their self-determination.

On Figure 3, the result of clustering the data into 3 clusters is shown. Comparing it with Figure 4, which shows the result of automatic selection of the number of clusters, we can see that the KAMILA algorithm is capable of choosing the cluster region incorrectly when the number of clusters is manually specified, and also this algorithm tends to merge two areas with a smaller data spread.

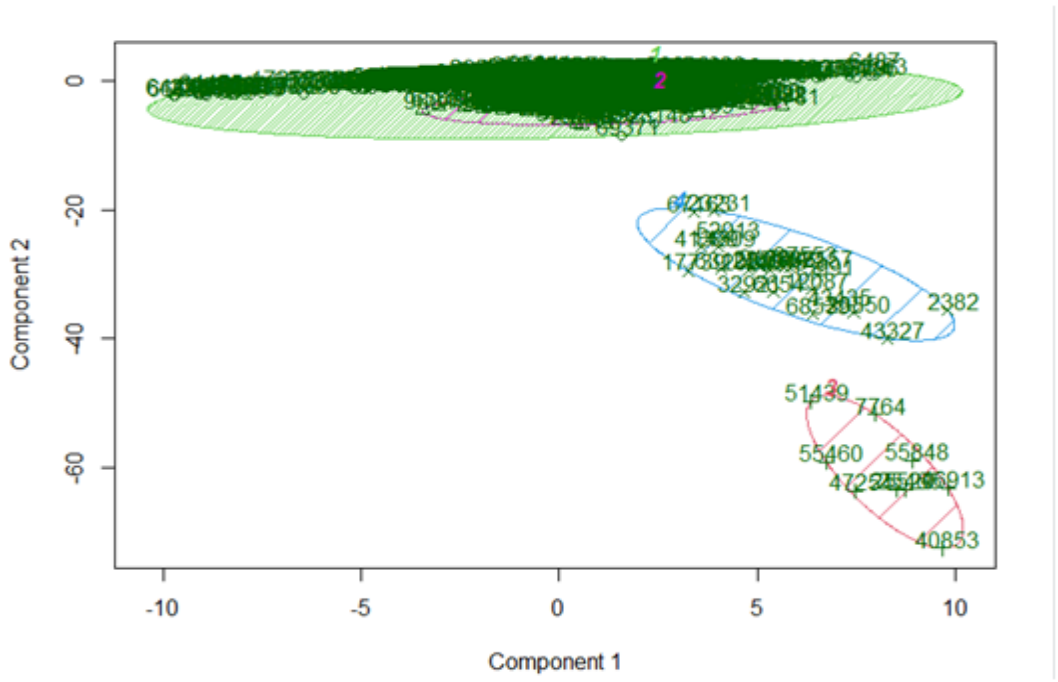


Figure 3. The result of clustering the dataset with 70,000 records into 3 clusters using the KAMILA algorithm.

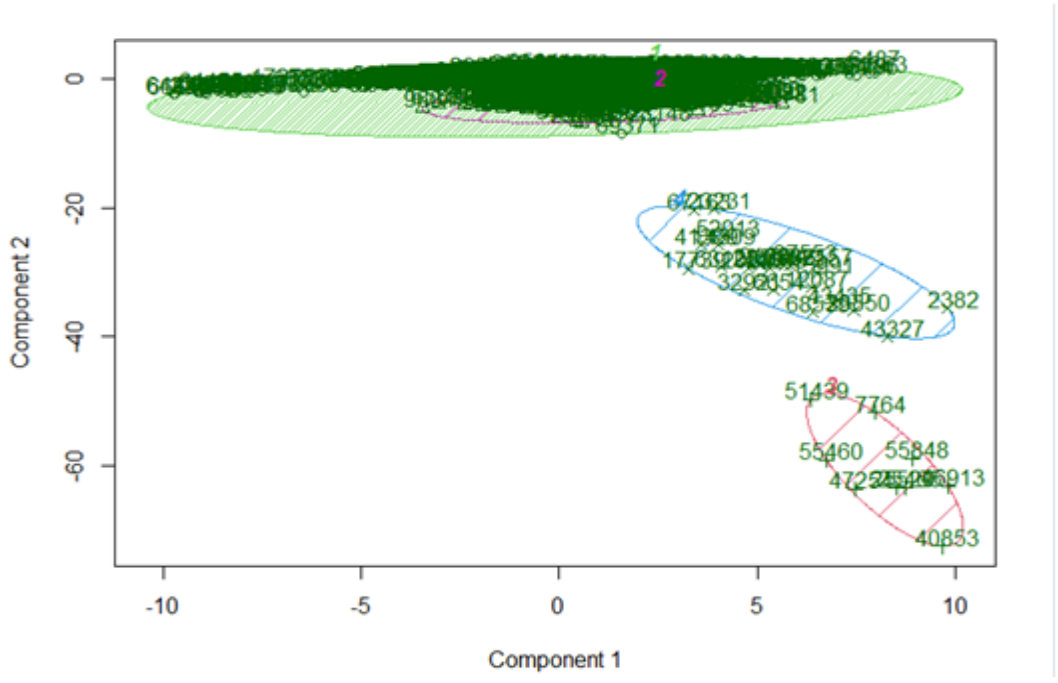


Figure 4. The result of clustering the dataset with 70,000 records using the KAMILA algorithm.

We will perform clustering by reducing the number of records by half. From Figure 5, we can see that clusters with small distances between features tend to merge when the number of records is reduced.

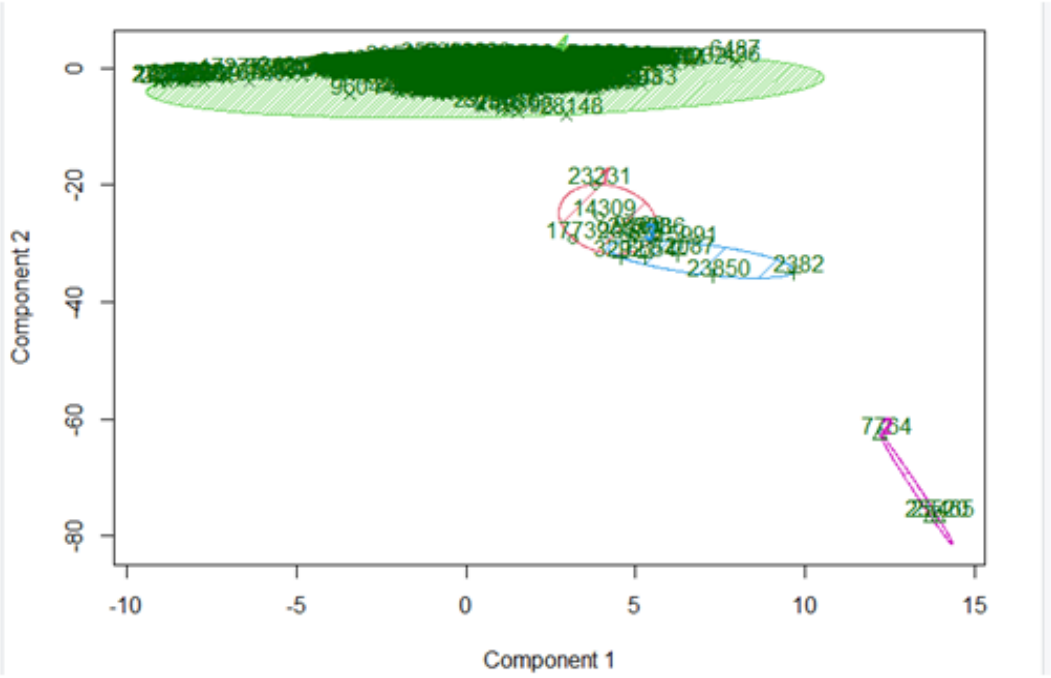


Figure 5. The result of clustering the dataset with 35,000 records using the KAMILA algorithm.

On Figures 6 and 7, the results of clustering using the k-prototypes algorithm are shown. Comparing the performance of the KAMILA and k-prototypes algorithms, it can be concluded that the k-prototypes algorithm is not able to accurately cluster the data compared to KAMILA, due to the error in weights of categorical data. However, the execution time of the KAMILA algorithm was 1.67 seconds, while the k-prototypes algorithm took 61.68 seconds.

As clustering of heterogeneous data differs significantly from traditional clustering, a hybrid distance measure for categorical and numerical values should be used to avoid certain limitations of traditional approaches.

Hybrid distance is a concept that combines each branch representing a certain feature with values and weight, which corresponds to the distance between values.

In Figure 8, a typical hierarchical concept is shown, consisting of edges and vertices, where higher-order vertices represent more general criteria, and lower-order vertices represent more specific features. To find the distance between two such vertices, the weight of the edges must be added. There are several ways to determine these weights, for example, one can first determine the distances between two child vertices, and then find the distance between higher-order vertices. The simplest way is to assign the same weight, for example 1, to all edges.

Hybrid distance with edge weight 1 is a type of distance metric used in clustering algorithms that combines different branches representing different features in a hierarchical tree structure. The weight of each edge represents the distance between two features. In the case of hybrid distance with edge weight 1, each edge is assigned a weight of 1, which makes all branches equally important in determining the overall distance between two observations.

The distance between two categorical values in the two models is the common

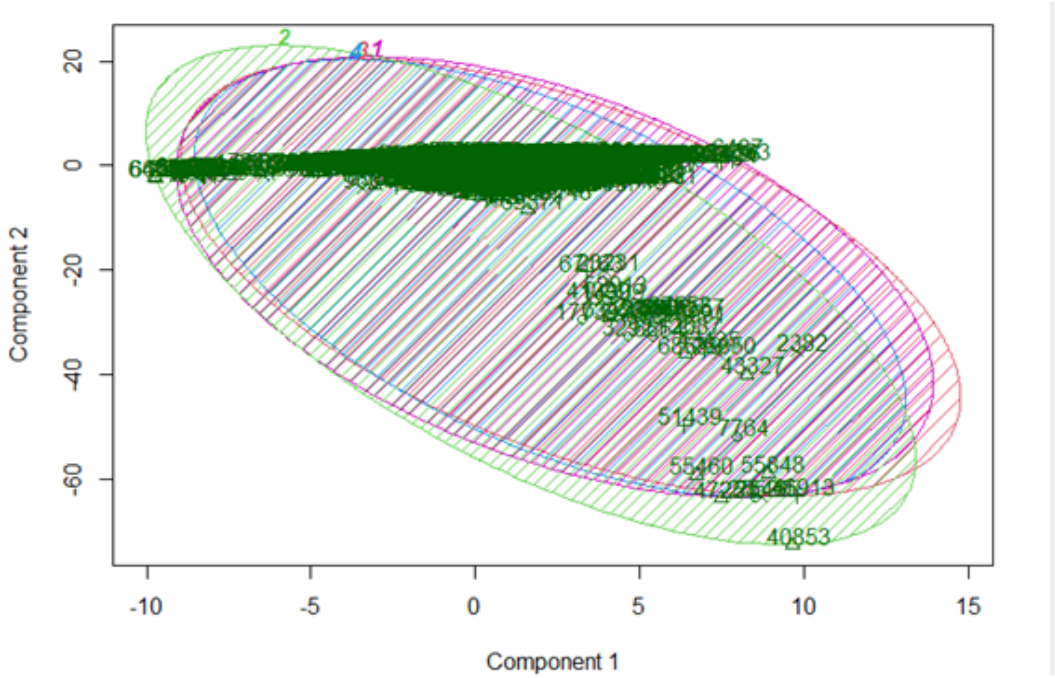


Figure 6. Result of clustering using k-prototypes algorithm on dataset with 70000 records.

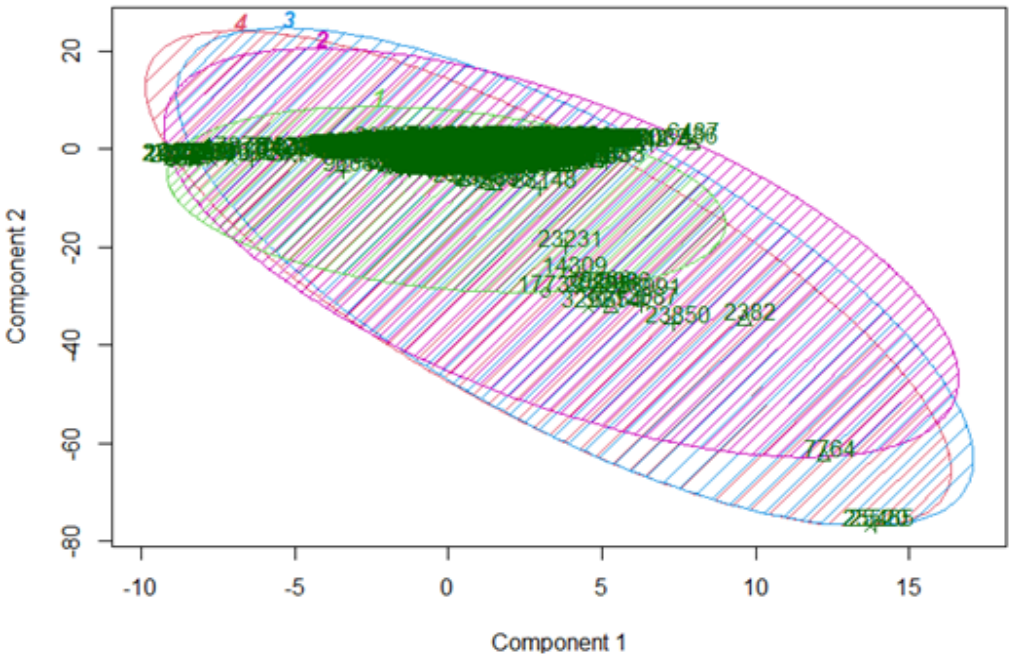


Figure 7. Result of clustering using k-prototypes algorithm on dataset with 35000 records.

weight of the path formed by two leaf nodes that represent the categorical values.

To perform hierarchical clustering, we need to calculate the Gower distance and the distance matrix. We will use the built-in libraries of the R language for this. Also, in order to see a visual result of hierarchical clustering, we need to reduce

our data to 50 rows. Once we are confident that hierarchical clustering has worked correctly, we will increase the data to obtain a more accurate result (Fig. 8).

```
1225 dissimilarities, summarized :
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
0.02863 0.21284 0.28658 0.29454 0.36564 0.68241
Metric :  mixed ;  Types = I, I, S, I, I, I, I, N, N, N, N, N, N
Number of objects : 50
```

Figure 8. General characteristics of the Gower distance.

From Figure 8, we can see that our metric has mixed data. Now we need to determine the number of clusters. To do this, we will apply the *Average Silhouette Width*.

```
$nc
[1] 2
```

Figure 9. Clusters quantity.

As can be seen, the algorithm split the described data into two clusters. Therefore, it is necessary to examine the partition more closely. The agglomerative clustering algorithm split each record in the dataset used into a specific cluster.

```
Clustering vector:
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
 1  2  2  2  1  1  1  2  1  1  1  1  1  1  1  2  1  1  1  1  1  1  2  1  2
26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50
 1  1  1  1  1  2  2  1  2  2  1  1  1  2  2  1  2  2  2  2  2  2  1  1  1
```

Figure 10. Clustering vector.

Initially, each branch forms its own cluster. Then, the closest clusters are identified and merged into one cluster. The merging process is repeated until all patterns form a single cluster. The output of hierarchical clustering is usually represented in the form of a dendrogram, as shown in Figure 10.

Now, regarding the latest calculations, let's display the hierarchical clustering graphs with different methods. The clustering algorithms vary in identifying the closest clusters for merging. The three most popular options are single linkage, complete linkage, and average linkage. The first approach measures the distance between two clusters by the minimum distance between any two points in these two clusters. In contrast, the complete linkage approach measures the distance by the maximum distance between any two points in these two clusters. For average linkage, the distance is measured by the average distance between two schemata of two clusters.

In the presented study, it is recommended to use two methods as it provides a more accurate dendrogram. The following features should be used for clustering: age, gender, height, weight of the patient, as well as whether the patient smokes, drinks alcohol, and their cardiovascular indicators: cholesterol, glucose, lower and upper blood pressure, and whether the patient does cardio exercise (see Figure 11).

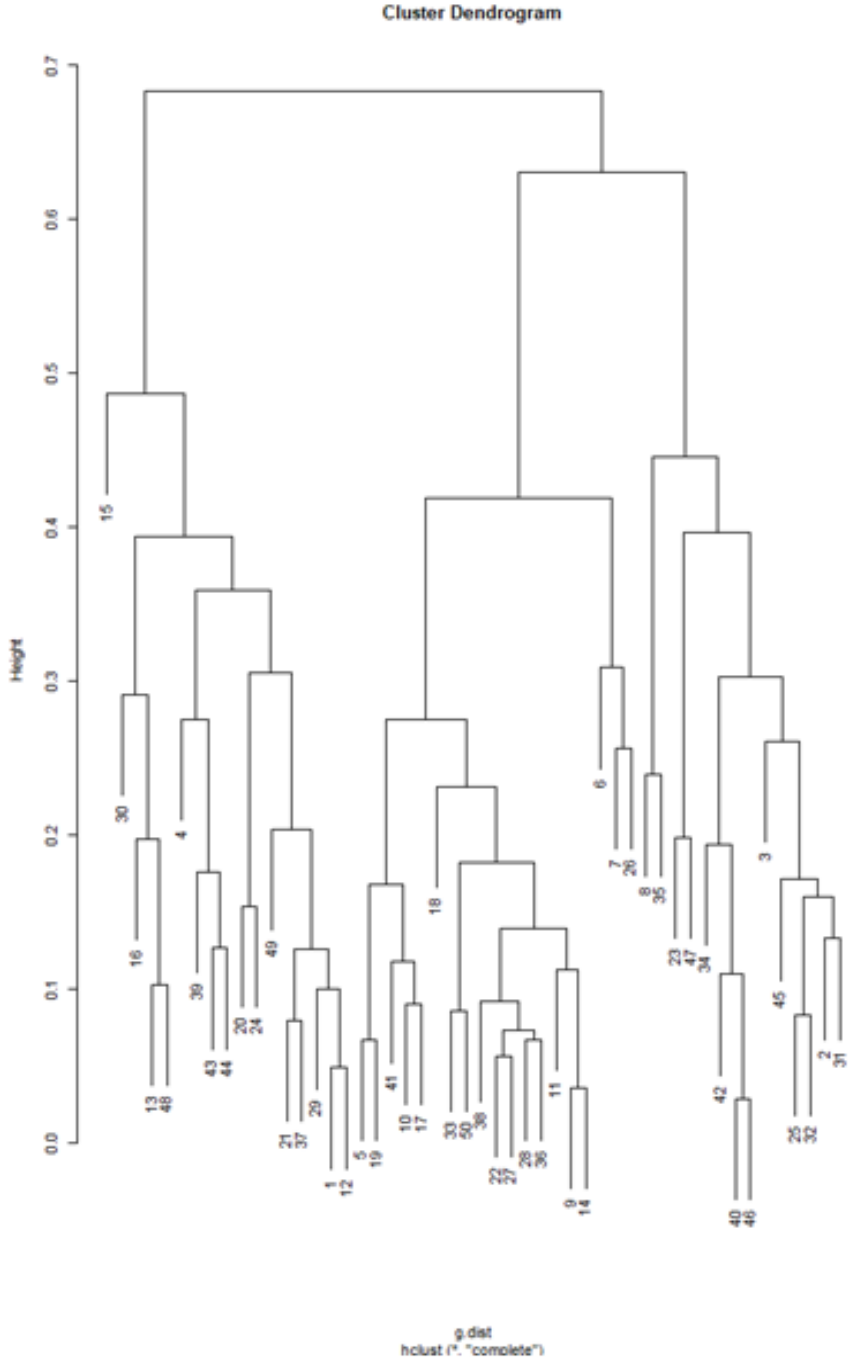


Figure 11. Results of hierarchical clustering using complete linkage method.

As expected, there are 2 clusters in the result. Let's analyze the 6th and 20th records as an example (Figure 12).

As can be seen from the dendrogram, hierarchical clustering assigned the sixth

| | age | gender | height | weight | ap_hi | ap_lo | cholesterol | gluc | smoke | alco | active | cardio |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 6 | 21914 | women | 151 | 67 | 120 | 80 | above normal | above normal | smokes | doesn't drink alco | 0 | 0 |
| 20 | 21755 | men | 162 | 56 | 120 | 70 | normal | normal | doesn't smoke | doesn't drink alco | 1 | 0 |

Figure 12. The 6th and 20th records from the patient dataset.

record to cluster 2 and the 20th record to cluster 1. These two records are in different clusters because their features differ significantly from each other. For example, in the two records, the gender, cholesterol, and glucose are different. Additionally, the person from the sixth record is a smoker and leads a sedentary lifestyle, whereas the person from the 20th record exercises regularly. The common feature between the two records is cardiovascular exercise.

Now that we have confirmed that the algorithm works correctly, let's increase the amount of data to 5000.

For this amount of data, the algorithm found 4 clusters, as shown in Figure 13.

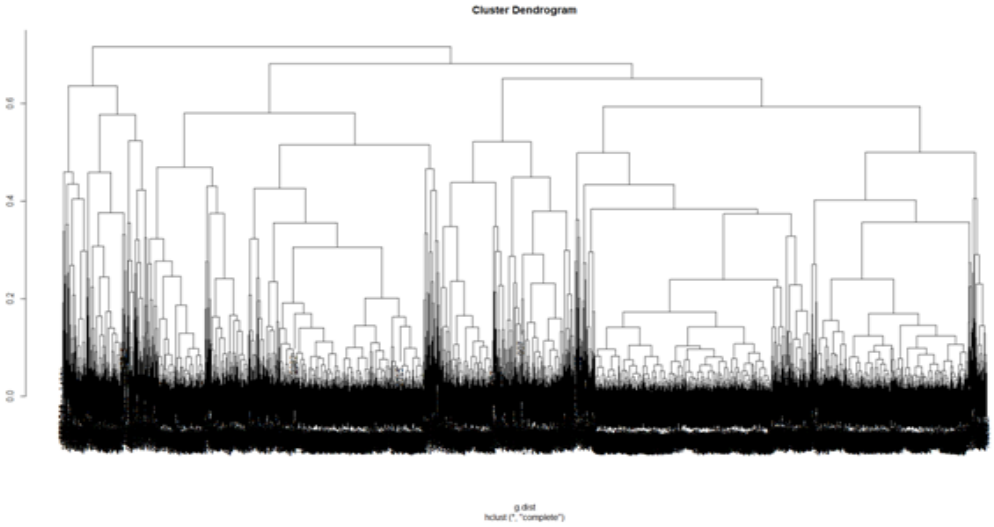


Figure 13. Amount of clusters for 5000 items.

Overall, we can see that the algorithm has divided our data into 4 clusters. Comparing hierarchical clustering with previous ones, we can say that it is quite accurate, but unfortunately it takes a lot of resources and time. It took 40 seconds to cluster 5000 data (Figure 14). If we were to cluster the entire dataset of 70000 data points, it would take up to 30 minutes, as the hierarchical clustering algorithm runs in $O(n^3)$ time and requires $\Omega(n^2)$ memory.

**5. Results.** During the experiments, there was a problem with presenting the results of the algorithms in a graphical format, which was due to the presence of 12 different features for clustering, 7 of which were categorical. The problem of representing data in a graphical format with 12 different features, 7 of which were categorical, was overcome by using the PCA algorithm to transform categorical features into numerical ones and reduce the dimensionality of the data to 2 components. This was followed by an orthogonal projection of the clusters onto the 2D space. The approach of using PCA to reduce dimensionality to 2 components and

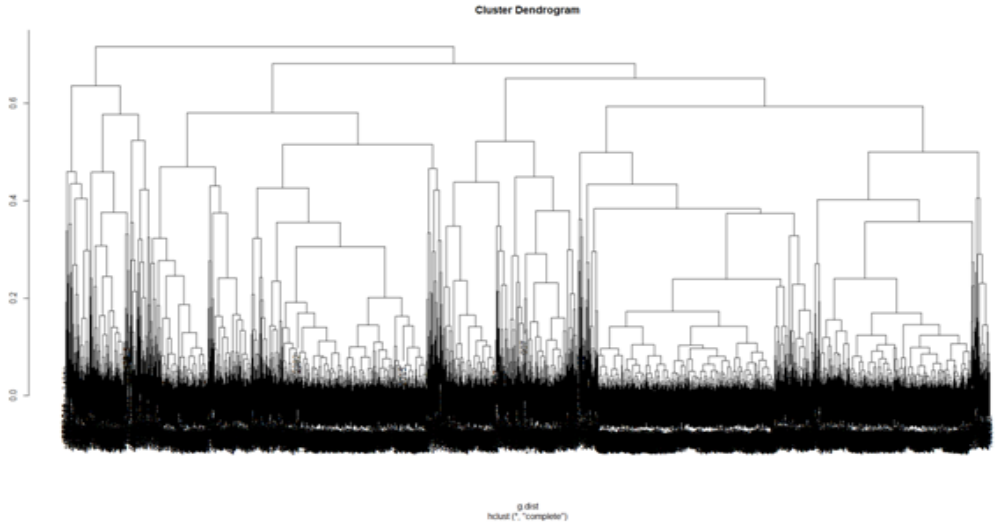


Figure 14. Results of hierarchical clustering using complete linkage method.

overlaying clusters on them helped to visualize the results of clustering algorithms, but it also resulted in significant information loss, as demonstrated by the example of applying k-prototypes clustering.

Conducting experiments on hierarchical clustering of heterogeneous data, we can note the advantages and disadvantages of this approach. Among the advantages are the ability to visually display the results of clustering using dendrograms without preprocessing the data, which avoids loss of information. Dendrograms provide the user with the ability to quickly identify typical representatives of dataset clusters. For example, using the cardio indicators dataset of hospital patients we were able to divide the patient records into 4 groups, which in practice can help in determining the approach to treating a particular patient. As an advantage of this algorithm, one can also mention the ability to quickly track the influence of different types of features on clusters and to find the distance between different categorical features. After conducting experiments, some disadvantages of hierarchical clustering of heterogeneous data can be noted, such as longer execution time compared to partitioning clustering algorithms such as K-Means and k-Prototypes, which can be explained by the quadratic complexity of hierarchical algorithms compared to linear complexity of partitioning ones. Taking into account these factors, as well as the complexity of presenting the results of analysis of large datasets, it can be noted that clustering of large datasets with mixed features is impractical, as the KAMILA algorithm implemented on the MapReduce distributed computing model will have a significant advantage in terms of speed.

Based on the experiments conducted with the k-prototypes algorithm, it can be noted that the main advantage of this method is its relative simplicity in implementation. However, in terms of other properties, the k-prototypes method lags behind the KAMILA algorithm. Some of the disadvantages of k-prototypes algorithm include the complexity of finding the weights between categorical and numerical data, which reduces the speed of the algorithm and leads to inaccurate results. Additionally, the random selection of cluster centers during initialization and the difficulty in determining the number of potential clusters make the results more variable. During

the experiment, it was observed that the accuracy of the algorithm was low, which was caused by the loss of information due to the use of the Hamming distance and the predominance of categorical data in the dataset.

The KAMILA method is a modern method of clustering heterogeneous data. Unlike the clustering algorithms demonstrated in the experiments, this algorithm is much more efficient. This algorithm is as fast as the usual k-means algorithm, but unlike this algorithm and other divisive clustering algorithms, it is much more robust to outliers, since KAMILA is a model-based algorithm using KDE. Overall, the KAMILA algorithm is a modern method for clustering heterogeneous data. Unlike the clustering algorithms demonstrated in the experiments, this algorithm is much more efficient. The algorithm is as fast as regular k-means, but unlike this algorithm and other divisive clustering algorithms, KAMILA is much more robust to outliers, as it is a model-based algorithm that uses KDE. The algorithm also accurately separates data into clusters and does not require a lot of memory during execution. Thanks to its characteristics, this algorithm can be used for clustering large datasets. Using the KAMILA algorithm is recommended when the distribution of the data is unknown.

The only significant disadvantage of this algorithm is that it can be difficult to understand. Additionally, KAMILA may produce less accurate results for data in a normal distribution.

The Average Silhouette method can be used for optimal selection of the number of clusters, which allows validating the membership of each point to a particular cluster. After running the clustering algorithm with a certain number of clusters, the silhouette analysis is applied to it, returning a number from -1 to 1. The closer this number is to 0, the closer the majority of points are to their center.

**Discussion of the results.** During the conducted research, general concepts of clustering and its methods were considered. As a result, it was examined and analyzed that mixed-type data are quite common in many fields, but there are no effective clustering strategies for such datasets. In other words, existing methods use arbitrary management strategies for clustering continuous and categorical features, which often leads to undesirable solutions dominated by one or the other type. The issues with conventional methods for clustering mixed data were characterized and it was confirmed that to use clustering methods for mixed data, it is necessary to choose a dataset with a moderate number of records so that these methods take less time, but it is also important to understand that accuracy cannot be achieved without some loss. In addition, it was understood that processed data is needed for the clustering of mixed-type data, meaning data that has been cleaned of unnecessary information and transformed into appropriate formats. The experiment also explored how to consider the similarity inherent in categorical values during hierarchical clustering, using the hierarchy of distances approach, which not only facilitates the expression of similarity but also combines several widely accepted approaches to processing categorical data.

Based on the experiments conducted, it is possible to draw conclusions about the usefulness of certain clustering algorithms for mixed-type data, depending on the required accuracy, knowledge of the dataset, its size, as well as the resources of the working station and the desired speed. Based on the analysis and experiments, it was found that hierarchical clustering was the most appropriate method

for processing the cardio patient data, despite the significant processing time. The resulting clusters were the most informative and convenient for end-users, such as hospital staff. The results demonstrate that the proposed hierarchical clustering approach can better reveal the structure of data similarity, especially when categorical attributes are involved and their values have varying degrees of similarity. The main disadvantage of hierarchical clustering is its slow algorithm, which is compensated by the absence of frequent changes in the dataset. Based on the experiments conducted, it can be concluded that the KAMILA algorithm is the most effective in meeting the requirements of both speed and accuracy. After analyzing the results of the k-prototypes algorithm, it should be noted that using this method is only advisable with significant knowledge in the subject area of the data, to properly weight the categorical and numerical data. It should also be noted that clustering mixed data using distances can result in information loss, and if accuracy is a top priority, it may be more appropriate to use clustering algorithms based on models or neural networks for clustering mixed data. In addition, it should be noted that the methods that were not used in this coursework were rejected due to their difficult implementation or lack of information about them.

**Summary.** In summary, it can be concluded that the problem of clustering mixed-type data remains challenging, and the methods investigated in this study have significant limitations. When working with a small dataset, the accuracy of the results was poor, while using a larger dataset led to higher accuracy but significantly increased computation time. It is important to carefully consider the characteristics of the dataset, the desired level of accuracy, and the available computational resources when choosing a clustering method for mixed-type data. Additionally, further research is needed to develop more efficient and accurate algorithms for clustering mixed-type data. The biggest problem is the lack of information about these methods in general. In most sources, the methods are characterized as a general concept, and experiments are almost always absent. In conclusion, it can be said that the field of clustering of heterogeneous data is not fully explored. Therefore, clustering of heterogeneous data is a challenging process that requires a lot of experience in this field and skills to implement one's work to improve existing algorithms or create new methods for clustering heterogeneous data.

## References

1. Sarker, A. (2018). Employee's performance analysis and prediction using K-means clustering & decision tree algorithm. *Global Journal of Computer Science and Technology.* 18(1), 1–5.
2. Fraley, C., & Raftery, A. E. (1998). How Many Clusters? Which Clustering Method? Answers Via Model-Based Cluster Analysis. *Technical Report,* 41(8), 578–588. https://doi.org/10.1093/comjnl/41.8.578
3. Murtagh, F. (2020). A survey of recent advances in hierarchical clustering algorithms which use cluster centers. *Computer Journal,* 26(4), 354–359. https://doi.org/10.1093/comjnl/26.4.354
4. Saxena, A., Prasad, M., Gupta, A., Bharill, N., Patel, O. P., Tiwari, A., & Lin, C. T. (2017). A review of clustering techniques and developments. *Neurocomputing,* 267, 664–681. https://doi.org/10.1016/j.neucom.2017.06.053
5. Sneath, P., & Sokal, R. (1973). Numerical Taxonomy. *Human Biology,* 47(2), 285–288.
6. Ptitsyn, A., Hulver, M., Cefalu, W., York, D., & Smith, S. R. (2006). Unsupervised clustering of gene expression data points at hypoxia as possible trigger for metabolic syndrome. *BMC Genomics,* 7(318), https://doi.org/10.1186/1471-2164-7-318
7. Tung, A. K., Hou, J., & Han, J. (2001). Spatial clustering in the presence of obstacles. Proceedings 17th International Conference on Data Engineering. Heidelberg.

https://doi.org/10.1109/ICDM.2002.1184042
8. Bohm, C., Railing, K., Kriegel, H., & Kroger, P. (2004). Density connected clustering with local subspace preferences. Proc. of the 4th IEEE Intern. conf. on data mining. Los Alamitos. https://doi.org/10.1007/978-0-387-39940-9_605
9. Boyko, N., Kmetyk-Podubinska, K., & Andrusiak, I. (2021). Application of Ensemble Methods of Strengthening in Search of Legal Information. *Lecture Notes on Data Engineering and Communications Technologies,* 77, 188–200. https://doi.org/10.1007/978-3-030-82014-5_13
10. Boyko, N., Hetman, S., & Kots, I. (2021). Comparison of Clustering Algorithms for Revenue and Cost Analysis. Proceedings of the 5th International Conference on Computational Linguistics and Intelligent Systems. Kharkiv [in Ukrainian].
11. Procopiuc, C. M., Jones, M., Agarwal, P. K., & Murali, T. M. (2002). A Monte Carlo algorithm for fast projective clustering. ACM SIGMOD Intern. conf. on management of data. Madison.
12. Boyko, N. (2016). Application of mathematical models for improvement of "cloud" data processes organization. *Mathematical Modeling and Computing,* 3(2), 111–119. https://doi.org/10.23939/mmc2016.02.111
13. Hossain, M. Z., Akhtar, M. N., Ahmad, R. B., & Rahman, M. (2017). A dynamic K-means clustering for data mining. *Indonesian Journal of Electrical Engineering and Computer Science,* 13(2), 521–526. https://doi.org/10.11591/ijeecs.v13.i2.pp521-526
14. Slamet, C., Rahman, A., Ramdhani, M. A., & Darmalaksana, W. (2016). Clustering the verses of the Holy Qur'an using K-means algorithm. *Asian Journal of Information Technology,* 15(24), 5159–5162.
15. Bekiros, S., Nguyen, D. K., Sandoval, J. L., & Uddin, G. S. (2017). Information diffusion, cluster formation and entropy-based network dynamics in equity and commodity markets. *European Journal of Operational Research,* 256(3), 945–961. https://doi.org/10.1016/j.ejor.2016.06.052

**Бойко Н. І., Ткачик О. А.** Алгоритми та методи кластеризації для різноманітних даних.

Дослідження присвячено комплексному вивченню методів кластеризації різнотипових даних. Досліджуються проблеми алгоритмів графічного формату, що зумовлені наявністю 12-ти різних ознак для кластеризації, 7 з яких були категоріальні. Представлене подання даних по 12-ти осях в графічному форматі. Було вирішено застосувати алгоритм PCA з перетворенням категоріальних ознак в числові для зменшення розмірності даних до 2-х компонент й подальшого ортогонального накладання кластерів на них. Наводиться застосування кластеризації методу к-прототипів. Показане використання PCA для зменшення розмірності в 6 разів приводить до значної втрати інформації. Проведені експерименти щодо ієрархічної кластеризації різнотипових даних, можна відзначити переваги й недоліки даного підходу. Наведена складність проведення кластеризації, яка полягає у представленні результатів аналізу великих даних. Описаний алгоритм KAMILA, який реалізований на моделі розподілених обчислень MapReduce і дає значну перевагу по швидкодії.

**Ключові слова:** максимізація очікування, моделювання структурних рівнянь, KAy-середні для даних MIxedLArge, найменший спільний предок, карта самоорганізації, теорія адаптивного резонансу, оцінка щільності ядра.

## Список використаної літератури

1. Sarker A. Employee's performance analysis and prediction using K-means clustering & decision tree algorithm. *Global Journal of Computer Science and Technology.* 2018. Vol. 18, No. 1. P. 1–6.
2. Fraley C., Raftery A. E. How Many Clusters? Which Clustering Method? Answers Via Model-Based Cluster Analysis. *Technical Report.* 1998. Vol. 41, No. 8. P. 578–588. DOI: https://doi.org/10.1093/comjnl/41.8.578
3. Murtagh F. A survey of recent advances in hierarchical clustering algorithms which use cluster centers. *Computer Journal.* 2020. Vol. 26, No. 4. P. 354–359. DOI: https://doi.org/10.1093/comjnl/26.4.354

4. 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. 267, 664–681. DOI: https://doi.org/10.1016/j.neucom.2017.06.053
5. Sneath P., Sokal R. Numerical Taxonomy. *Human Biology.* 1973. Vol. 47, No. 2. P. 285–288.
6. 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.* 2006. Vol. 7, No. 318. https://doi.org/10.1186/1471-2164-7-318
7. Tung A. K., Hou J., Han J. Spatial clustering in the presence of obstacles. Proceedings 17th International Conference on Data Engineering. Heidelberg, 02–06 April 2001. Heidelberg, 2001. P. 359–367. DOI: https://doi.org/10.1109/ICDM.2002.1184042
8. Bohm C., Railing K., Kriegel H., Kroger P. Density connected clustering with local subspace preferences. Proc. of the 4th IEEE Intern. conf. on data mining. Los Alamitos, 01 November 2004. Los Alamitos, 2004. P. 27–34. DOI: https://doi.org/10.1007/978-0-387-39940-9_605
9. 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. P. 188–200. DOI: https://doi.org/10.1007/978-3-030-82014-5_13
10. 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. Kharkiv, 22–23 April 2021. Kharkiv, 2021. P. 1866–1877.
11. 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, 03 June 2002. Madison, 2002. P. 418–427. https://doi.org/10.1145/564691.564739
12. Boyko N. Application of mathematical models for improvement of "cloud" data processes organization. *Mathematical Modeling and Computing.* 2016. Vol. 3, No. 2. P. 111–119. https://doi.org/10.23939/mmc2016.02.111
13. Hossain M. Z., Akhtar M. N., Ahmad R. B., Rahman M. A dynamic K-means clustering for data mining. *Indonesian Journal of Electrical Engineering and Computer Science.* 2017. Vol. 13, No. 2. P. 21–526. https://doi.org/10.11591/ijeecs.v13.i2.pp521-526
14. Slamet C., Rahman A., Ramdhani M. A., Darmalaksana W. Clustering the verses of the Holy Qur'an using K-means algorithm. *Asian Journal of Information Technology.* 2016. Vol. 15, No. 24. P. 5159–5162.
15. Bekiros S., Nguyen D. K., Sandoval J. L., Uddin G. S. Information diffusion, cluster formation and entropy-based network dynamics in equity and commodity markets. *European Journal of Operational Research.* 2017. Vol. 256, No. 3. P. 945–961. https://doi.org/10.1016/j.ejor.2016.06.052

Одержано 16.02.2023