Data segmentation through two-level clustering with greedy approach

Vladimir Kazakovtsev[1,2], and *Egor* Markushin[1]

[1]Reshetnev Siberian State University of Science and Technology, 31, Krasnoyarskii rabochii prospekt, Krasnoyarsk 660037, Russian Federation
[2]Siberian Federal University, 79 Svobodny Prospekt, Krasnoyarsk 660041, Russian Federation

Abstract. This study presents a two-level clustering method utilizing a simplified greedy procedure to enhance data processing efficiency and accuracy, particularly with large high-dimensional datasets. The two-level structure allows for the identification of broad data groups in the first stage, followed by a more granular analysis within these groups in the second stage, thereby accelerating the clustering process and improving result quality. The application of the k-means++ method did not yield the anticipated benefits compared to traditional random initialization. Such findings underscore the necessity for preliminary data analysis when selecting optimal clustering algorithms, as instances of complex methods failing to improve results are not uncommon. This work illustrates the importance of balance between method complexity and effectiveness in real-world applications and emphasizes the potential for increased resource expenditure without commensurate gains in clustering performance.

1 Introduction

Clustering is a fundamental technique in data analysis and machine learning that involves grouping a set of objects, data points, or observations into clusters based on their estimated similarities or characteristics. The primary goal of clustering is to maximize the inner-cluster similarity while minimizing the inter-cluster similarity, ultimately identifying structures and patterns that may not be immediately apparent in the data. This unsupervised learning method can be valuable in various fields, including market research, biology, social science, and image analysis, as it helps summarize vast amounts of information, enabling better decision-making and revealing insights that can drive strategic actions.

At its core, clustering algorithms seek to partition datasets into meaningful groups. Each group, or cluster, contains items that share common features, making them more similar to one another than to those in different clusters. The critical aspect of clustering lies in its ability to discern these hidden patterns without prior labels or classifications, making it especially useful in exploratory data analysis.

There are several clustering algorithms, each with its own strengths and weaknesses. Some popular methods include K-means clustering, hierarchical clustering, density-based algorithms, such as DBSCAN, Gaussian mixture models and others. These techniques can be applied across various data types, whether the data is numerical, categorical, or in high-dimensional spaces. The choice of algorithm often depends on the specific characteristics of

the dataset, the desired outcome, and the assumptions that can be made about the underlying structure of the data.

As datasets grow larger and more diverse, existing clustering algorithms need adaptation to maintain quality and speed. Traditional algorithms analyze all dimensions of a dataset, but in high-dimensional data, many dimensions may be irrelevant, leading to confusion and masking of clusters. In very high dimensions, data points often become nearly equidistant, obscuring clusters. While feature selection methods have been somewhat successful in improving cluster quality by removing irrelevant dimensions, subspace clustering algorithms offer a more focused approach. They search for and uncover clusters within multiple, possibly overlapping subspaces rather than examining the entire dataset, but also have certain disadvantages [1].

Clustering not only aids in the organizing of data but also facilitates pattern recognition, anomaly detection, and predictive modeling. Thus, in customer segmentation tasks, businesses can use clustering to identify distinct groups of customers based on purchasing behavior, enabling targeted marketing strategies. In healthcare, clustering can help in identifying disease subtypes based on patient symptoms and genetics, leading to more personalized treatments.

Overall, clustering is a powerful tool for simplifying complex datasets, revealing unseen relationships, and supporting a myriad of applications in research and industry. Its ability to categorize and interpret data relationally makes it an indispensable technique in the modern data-driven landscape.

2 Clustering algorithms

Let there be a set of input data $X = \{x_1, \ldots, x_j, \ldots, x_N\}$, where $x_j = \left(x_{j1}, x_{j2}, \ldots, x_{jd}\right)^T \in \mathbb{R}^d$. Two clustering methods for X can be distinguished:

The *hierarchical clustering* algorithm attempts to build a division of the data X in the form of a nested tree structure $H = \{H_1, \ldots, H_Q\}$, $(Q \leq N)$, such that $C_i \in H_m, C_j \in H_l, m > l$ means that $C_i \in C_j$ or $C_i \cap C_j = \emptyset$ for all $j, i \neq j, m, l = 1, \ldots, Q$ [2].

The non-hierarchical (flat) clustering algorithm attempts to divide X into K groups $C = \{C_1, \ldots, C_K\}, (K \leqslant N)$ such that

$C_i \neq \emptyset, i = 1, \ldots, K$;

$\bigcup_{i=1}^{K} C_i = X$;

$C_i \cap C_j = \emptyset, i, j = 1, \ldots, K$ and $i \neq j$.

To date, there are dozens of cluster analysis methods that allow the identification of groups with various parameter combinations, the most well-known and frequently used of which is the k-means method. As a local optimization algorithm, *k*-means is dependent on the choice of initial values (the initial positions of the centroids), but it possesses reproducibility: when the same initial positions of centroids are chosen, the algorithm will yield the same result, which is an undeniable advantage for certain classes of tasks, including manufacturing tasks.

Similar to the p-median problem [2], the k-means problem is a classical location problem: it requires finding k cluster centers $X_1, \ldots, X_k$ in d-dimensional space that minimize the sum of the squared distances from these centers to given points A_i.

$$F(X_1, \ldots, X_k) = \arg\min_{X \in \Re^d} \sum_{i-1}^{N} \min_{X \in \{X_1, \ldots, X_k\}} \|A_i - X\|^2$$

A classical method for solving the k-means problem is the eponymous algorithm, also known as the ALA procedure (Alternating Location-Allocation) or Lloyd's algorithm [3] (Algorithm 1), named after its creator. The algorithm consists of just two alternating steps:

1. *Group Assignment*: Assign points to clusters around known centers (an object belongs to the group whose center is the nearest to it).

2. *Centroid Update*: The new centroid becomes the center of mass of the objects in the clusters.

The algorithm seeks a local minimum by iteratively improving a known solution. However, it has certain limitations; in particular, the number of groups k into which objects are to be divided must be specified in advance. The result is highly dependent on the initial solution, which is usually chosen randomly. To find a global minimum, a multi-start approach for the k-means algorithm is used, or various global optimization methods that incorporate the k-means algorithm are employed.

Algorithm 1 k-means or Lloyd Procedure

Require: Set of initial centers $S = \{X_1, ..., X_k\}$. If S is not given, then the initial centers are selected randomly from the set of data vectors $\{A_1, ..., A_N\}$.

repeat

Step 1: For each center X_j , $j = 1,\ldots,k$, a subset G_j of data vectors for which X_j is the nearest center;

Step 2: For each subset G_j , $j = 1,\ldots, k$, calculate its new center having solved the Weber problem with Weiszfeld algorithm on G_j (in the case of p-median problem), or having averaged all data vectors in G_j (in the case of k-means problem);

until all centers stay unchanged.

Greedy agglomerative clustering [4] is a clustering method that operates in a bottom-up fashion. It starts from a local solution with extensive number of clusters and progressively merges the clusters until a specified number of clusters is reached. At each iteration, the algorithm removes the cluster whose removal results in the smallest increase in the objective function. The process involves calculating distances between clusters using various metrics (e.g. Euclidean or Manhattan). This method is effective for discovering relationships within data and can be visualized using dendrograms to illustrate the clustering process (Algorithm 2).

Algorithm 2 Basic Agglomerative search

Require: Set of initial centers $S = \{X_1,...,X_K\}$, $K > k$, required number k.
$S \leftarrow$ Lloyd(S);
while $|S| > p$ **do**
  for i=1,…, K **do**
    $F_i \leftarrow F(S\backslash\{X_i\})$;
  end for
  Select a subset $S' \subset S$ of r_{elim} centers with the minimum values of the corresponding variables F_i; $r_{elim} = 1+0.25(K-k)$.
  $S \leftarrow$ Lloyd($S\backslash S'$);
end while

Greedy agglomerative algorithms have proven to be an effective tool for clustering. However, one of the main drawbacks of these methods is their high computational cost, which limits their application on large datasets.

To address this issue, a simplified agglomerative procedure is proposed for handling large datasets. This approach can reduce computational expenses while maintaining an acceptable quality of clustering (Algorithm 3).

Algorithm 3 Simplified Agglomerative search

Require: Two sets of centers: S_1, S_2.
$S_1 \leftarrow Lloyd(S_1)$;
$S_2 \leftarrow Lloyd(S_2)$
return $BasicAggl$ $(S_1 \cup S_2)$

Another important part of this research is the two-layer cluster structure [5, 6]. Two-level clustering allows for more efficient processing of large datasets through a structure consisting of two stages. In the first stage, global clustering is performed, where the data is divided into several large groups using various methods, such as K-means or hierarchical clustering. This step helps to significantly reduce the volume of analyzed data while preserving important structural patterns and dependencies that may be hidden in more complex datasets.

In the second stage of the process, local clustering is carried out, where each of the previously identified large groups is analyzed in more detail. The task here is to identify smaller subgroups or clusters, which allows for a deeper understanding of the internal structure of the data. This analysis may involve using the same methods as in the first stage, or it may prefer other techniques that can work more effectively with smaller volumes of data and the specific nature of the information within the clusters.

Our implementation of two-layer clustering is presented as Algorithms 4-6.

Algorithm 4 Level-2 centroid initialization

for l=1…k_1 do
  Assign $C''_l=C_l$; $I_l=l$;
end for;
for l=(k_1+1)…k_2 do
  Select randomly $i^* \in 1…N$ such that $A_{i^*} \neq C''_j$ $\forall j$=1…(l –1);
  For A_i, find its closest level-1 centroid C_{j^*}, j^*=1…k_1;
  Assign $I_l=j^*$;
end for;
Assign $n_{current}$=0;
for j=1…k_1 do
  Assign $B_j=n_{current}$;
  for l=1…k_2 do
    if $I_l=j$ **then**
      Assign $C'_{ncurrent}= C''_l$; $n_{current}=n_{current}$+1;
    end if;
  end for;
end for;
Assign $B_{k1+1} = n_{current}$.

Algorithm 5 Level-2 k-means clustering

while N_{chg} **do**
  Assign N_{chg}=false;
  Assign S_l=0 i=1…k_2;

Assign n_l=0 i=1…k_2;
for i=1…N **do**
For A_i, find its closest level-1 centroid C_{j^*}, j^*=1…k_1;
For A_i, find its closest level-2 centroid C'_{l^*} among l^*=B_{j^*}…B_{j^*+1};
if ml^* **then**
assign N_{chg}=true
end if;
Assign n_{l^*}=n_{l^*}+1; S_{l^*}=S_{l^*}+A_i;
end for;
for k=1…k_2 **do**
C'_j=S_j/n_j;
end for;
end while

Algorithm 6 Eliminating the level-2 centroids

Require: number of level-2 centroids to be eliminated r_{elim};
Assign r_l=0 j=1…k_2;
for each i=1…N **do**
Find the level-1 centroid which is closest to A_i. Let it be C_{j^*};
Among B_{j^*}… B_{j^*}, find the index l^* of the level-2 centroid C'_{l^*} which is closest to A_i;
Among B_{j^*}… B_{j^*}, find the index l^{**} of the level-2 centroid $C'_{l^{**}}$ which is the second closest closest to A_i;
Assign r_{l^*}=r_{l^*}+$||C'_{l^{**}}-A_i||^2$ – $||C'_{l^*}-A_i||^2$;
end for;
Form an array I'' of level-2 centroid indexes I_1…I_{k2} sorted by the corresponding values of r_l, descending;
for j=1…n_{elim} **do**
for l=j…(k_2 – 1) **do**
Assign C'_l=C'_{l+1};
for b=1…k_1+1 **do**
if $B_b \geq l$ **then**
assign B_b=B_{b-1}
end if;
end for;
end for;
end for;
Assign k_2=k_2-n_{elim}.

3 Numreical experiment

For the experiments, a high-dimensional SIFT dataset (128 features) was used. Clustering was performed using the p-median model (Euclidean metric) and the k-means model (squared Euclidean distance). Two initialization methods were also employed: random initialization of the initial solution and k-means++. The results of the computational experiments are presented in Tables 1-2.

Table 1. Comparative results of the clustering algorithms for the k-means problem.

Clustering algorithm	**Initialization Method**	**Achieved objective function value, 30 runs**		
		Max (worst)	**average**	**Min (best)**
1000000 data points, 256 centroids				
ALA	Random	23508792	23516733.6	23525960
ALA	k-means++	23500268	23513930	23521506
2-Layer Clustering	Random	23475608	23481650	23520708
2-Layer Clustering	k-means++	23475712	23478027.71	23481328
1000000 data points, 256 centroids				
ALA	Random	21898272	21905756	21913664
ALA	k-means++	21893030	21902027	21915006
2-Layer Clustering	Random	21771792	21774277	21776486
2-Layer Clustering	k-means++	21787168	21789126	21793534

Table 2. Comparative results of the clustering algorithms for the *p*-median problem.

Clustering algorithm	**Initialization Method**	**Achieved objective function value, 30 runs**		
		Max (worst)	**average**	**Min (best)**
1000000 data points, 256 centroids				
ALA	Random	1526660.5	1526894.047	1527368
ALA	k-means++	1526220	1526647.51	1526971
2-Layer Clustering	Random	1525145.75	1525279.839	1525429
2-Layer Clustering	k-means++	1525222	1525350.163	1525510
1000000 data points, 256 centroids				
ALA	Random	1472318	1472596	1473034
ALA	k-means++	1472276	1472479	1472652
2-Layer Clustering	Random	1467882	1467984	1468104
2-Layer Clustering	k-means++	1468462	1468553	1468695

The results of the experiments demonstrated positive dynamics, with a noticeable improvement in the values of the objective function compared to standard approach, indicating the high effectiveness of the proposed clustering strategy. However, despite the

theoretical advantages of the k-means++ method, in this case, it did not show significant improvements compared to random initialization. This suggests that for the specific nature of the data, random initialization can also be an adequate and effective approach, providing stable results.

4 Conclusion

During the conducted research, a two-level clustering method was applied using a simplified greedy procedure, which allowed for significant improvements in data processing. This approach provides deeper and more accurate segmentation of information, which is especially important when working with large high-dimensional data sets. The two-level structure of clustering enables the identification of large groups of data at the first level, followed by a more detailed analysis and distribution of elements within these groups at the second level. This not only accelerates the clustering process but also enhances the quality of the results.

It is also worth noting that, despite the theoretical advantages, the k-means++ method did not show improvements compared to the regular random initialization of clusters. In our experiment, k-means++ was unable to demonstrate the expected effectiveness, which may indicate that the specifics of the data or its structure do not contribute to a more successful initialization of cluster centers. This observation highlights the importance of preliminary data analysis and their characteristics for selecting the optimal algorithm, as instances where more complex methods do not lead to improved results are not uncommon. Nevertheless, these methods are still used for real-world tasks due to assumptions about their effectiveness. Such situations lead not only to increased computational resource expenditure but often to a degradation of the resulting output.

The two-level clustering based on the greedy procedure has its advantages in working with large volumes of high-dimensional data. This method not only allows for savings in computational resources but also effectively manages the complexity of the data, providing high speed and accuracy in clustering. In the context of rapidly increasing data volumes and the need for processing, simplified and adaptive algorithms are becoming key tools for analysts and researchers.

The work was supported by the Ministry of Science and Higher Education of the Russian Federation (Grant No. 075-15-2022-1121).

References

1. L. Parsons, E. Haque, H. Liu, *Subspace clustering for high dimensional data: a review* ACM sigkdd explorations newsletter **6**, 1 (2004).
2. P. Hansen, N. Mladenović, Variable neighborhood search for the p-median. Location Science, **5**, 4 (1997).
3. G. Hamerly, J. Drake, Accelerating Lloyd's algorithm for k-means clustering. Partitional clustering algorithms (2015).
4. H. S. Amarilies et al, IOP Conf. Ser.: Mater. Sci. Eng. **847** 012007 (2020).
5. W. Jia, Y. Tan, L. Liu, J. Li, H. Zhang, K. Zhao, *Hierarchical prediction based on two-level Gaussian mixture model clustering for bike-sharing system.* Knowledge-Based Systems, **178** (2019).
6. G. Cabanes, Y. Bennani. *A simultaneous two-level clustering algorithm for automatic model selection.* In Sixth International Conference on Machine Learning and Applications (2007).