MAFIA: An Adaptive Grid-Based Subspace Clustering Approach

Henrik Daniel Christensen[hench13@student.sdu.dk]

University of Southern Denmark, SDU
Department of Mathematics and Computer Science

Abstract. The abstract should briefly summarize the contents of the paper in 150–250 words.

Keywords: High-Dimensional Subspace Clustering · MAFIA · Critical Review · Comparative Study.

1 Introduction

Clustering is one of the main techniques within data mining. This technique is a descriptive method that tries to discover unknown patterns within a data set, by partitioning the data objects into subsets (*clusters*). Here, each object in a cluster is similar to one another, but different from objects in other clusters. Clustering is widely used in many applications, such as biology, web search and business intelligence [3, p. 444].

A simple clustering example is in the context of customer data, where it is useful to group similar customers together for the purpose of targeted advertising or placement of products within a store [4, p. 5].

The task of clustering data is challenging task, first of all as the data sets typical is large in size, which means that the clustering algorithm must be *scalable*. Additionally, data sets often contains numerous features (*attributes*), which introduces the problem of *curse of dimensionality*, which refers to a several challenges related to high-dimensional data spaces:

First, there is the issue of *concentration of distances*, a phenomenon in high-dimensional spaces where the distances between objects become increasingly similar as dimensionality increases. This means that data points tend to become nearly equidistant from one another, making it difficult for traditional distance-based algorithms to discover clusters.

Secondly, there is the problem of *local feature relevance* and *local feature correlation*, where only a subset of features or different combinations of feature correlations may be relevant for clustering. Consequently, feature reduction techniques like Principal Component Analysis (PCA), which project the original space onto a lower-dimensional subspace, are inadequate because they typically identify only one global subspace that best fits the entire dataset. Similarly, clustering algorithms that evaluate the entire feature space, such as DBSCAN, struggle to address this issue effectively. [4, p. 43–46]

Instead of relying on a global approach to feature selection, a local approach that addresses the issues of local feature relevance and local feature correlation is necessary. However, when clustering high-dimensional data we encounter two separate problems, which, however, both needs to be solved simultaneously. First, is the problem of finding the relevant subspaces of each cluster. Second, is the problem of finding the clusters in each relevant subspace. In the first problem, notice that the search space is in general infinite and for the second problem, to find the best partitioning of the objects is NP-complete. To solve them simultaneously, we need to employ heuristics into the clustering algorithms. [4, p. 6–7]

For many applications, it is reasonable to focus only on clusters in axis-parallel subspaces, thus restricting the search space to $O(2^d)$ dimensions. These algorithms are often called *projected clustering* or *subspace clustering* algorithms. Furthermore, these can be divided into two categories: *top-down*- and *bottom-up* approaches. In the top-down approach, the relevant subspaces for the clusters are determined starting from the full-space either using the so called *locality assumption* or using a random sampling. In contrast, bottom-up approaches, finds the relevant subspaces for the clusters from the original space starting from one-dimensional using the *downward closure property* (also called monotonicity property), that says, *if a subspace contains a cluster, then a superspace must also contain a cluster*, which can be used to prune (exclude) subspaces. [4, p. 8, 11]

1.1 Contributions

This paper examines three different bottom-up subspace clustering algorithms, with a primary focus on the MAFIA algorithm [6], a grid-based method that partitions the data space into adaptive grids using histograms. Since MAFIA extends the pioneering subspace clustering algorithm called CLIQUE [1], a comparative analysis between the two will be conducted. Since the two algorithms both are being grid-based, which may limit clusters to polygonal shapes. Also, the more flexible SUBCLU algorithm [5], which uses density-connected sets to allow clusters with arbitrary shapes, will also be analyzed and evaluated.

The remainder of the paper is organized as follows. In Section 2, first the two grid-based algorithms CLIQUE and MAFIA are analyzed and how they relate. Additionally, the density-based algorithm approach of SUBCLU will be analyzed and how it differ from the grid-based approach, as well as how it differ from the full-dimensional density-based approach of the well-known DBSCAN algorithm [2]. Section 3 gives a more detailed description of the MAFIA algorithm. Section 4 evaluate the performance of the three algorithms in terms of scalability, considering both data dimensionality and cluster dimensionality, as well as their overall cluster quality. Section 5 discusses the pros and cons of the three algorithms. Finally, Section 6 draws conclusions and points out future work.

2 Subspace Clustering Algorithms

2.1 Grid-based approach

Description of CLIQUE and briefly introduce how MAFIA extends it.

2.2 Density-based approach

Description of SUBCLU and describe how it relates to DBSCAN.

3 MAFIA

3.1 Adaptive Grids

3.2 MAFIA Algorithm

- Simplified version of algorithm

3.3 Candidate Dense Units (CDUs)

- Why "any" dense unit

4 Evaluation

The evaluation of the clustering algorithms was performed on a Intel i7 1.70 GHz processor (12th gen.) with 16 GB of RAM running Windows 11.

The evaluation of MAFIA was performed using GPUMAFIA, which was installed on a virtual machine using VirtualBox. The virtual machine was configured with 4 CPUs and 4 GB of RAM. The operating system was Ubuntu 24.04.1 LTS. In contrast, CLIQUE and SUBCLU were evaluated on the main machine (host) using ELKI. Thus, one should be careful to compare the results of MAFIA with those of CLIQUE and SUBCLU, as the execution environment may affect the results, however, their growth rate can be compared.

The evaluation project can be found at: https://github.com/henrikdchristensen/SDU-Data-Mining-Exam, where additional tests can be found as well a more detailed description of how to generate the data sets used in this project.

4.1 Data set generation

Two different data generation tools were used to create clustered datasets. The first is MDCGen, which is very efficient for large-scale, high-dimensional data using MATLAB.

The other used was *Artificial Cluster* using Java, which is more flexible in terms of cluster shapes (e.g., creation of Bezier curves). However, note that I have not been successful in generating datasets larger than 1 million data points with this program. So, for most cases, MDCGen is used.

4.2 Experimental Results

Clustering Quality 2 Clusters = OK 4 CLusters = NOT DETECTED.

Scalability with Data Set Size

Scalability with Data Dimensionality and Cluster Dimensionality I was not able to reproduce the results found in the Subclu article where little information about the synthetic data set is provided. Using the same parameters for all the tests seems a little weird as the behaviour of the algorithm is completly different as well as its performance. Furthermore, I could not use a data set with 100,000 data points, so for testing Subclu 20,000 points is suggested.

Real Data Sets

5 Discussion

- Discussion of the different approaches
- Discussion of results
- Limitations and Strengths
- Model: Input parameters; Assumptions on number, size, and shape of clusters; Noise
- Determinism
- Independence w.r.t. order of objects/attributes
- Assumptions on overlap/non-overlap of clusters/subspaces
- Effiency

Interpretability and usability: Clustering results should be easy to interpret and algorithms should produce clusters that are meaningful and comprehensible, making the results practical for decision-making.

Discovery of arbitrary-shaped clusters: Many clustering algorithms are limited to detecting spherical clusters based on distance measures (e.g. Euclidean distance). However, clusters in real-world data often take arbitrary shapes.

Minimize dependence on domain knowledge: Many clustering algorithms require users to provide specific input parameters, such as the number of clusters, which can be challenging to determine a prior. Reducing such parameters not only simplifies the process for users but also improve the reliability of the results.

Robust to noisy data: Real-world data is often noisy or contains outliers, which can distort clustering results. Clustering algorithms should be robust enough to handle noisy data, missing values, and outliers without degrading the quality of the clusters.

[3, p. 446-447]

6 Conclusion

Main findings

References

1. Agrawal, R., Gehrke, J., Gunopulos, D., Raghavan, P.: Automatic subspace clustering of high dimensional data for data mining applications p. 94–105 (1998). https://doi.org/10.1145/276304.276314, https://doi.org/10.1145/276304.276314
2. Ester, M., Kriegel, H.P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining. p. 226–231. KDD'96, AAAI Press (1996)
3. Han, J., Kamber, M., Pei, J.: Data Mining: Concepts and techniques. Elsevier (6 2011)
4. Kriegel, H.P., Kröger, P., Zimek, A.: Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering. ACM Trans. Knowl. Discov. Data **3**(1) (Mar 2009). https://doi.org/10.1145/1497577.1497578, https://doi.org/10.1145/1497577.1497578
5. Kröger, P., Kriegel, H.P., Kailing, K.: Density-connected subspace clustering for high-dimensional data **246-257** (04 2004). https://doi.org/10.1137/1.9781611972740.23
6. Nagesh, H., Goil, S., Choud, A., Choudhary, P.: Adaptive grids for clustering massive data sets (01 2002). https://doi.org/10.1137/1.9781611972719.7