Subspace clustering with gravitation *

Jiwu Zhao
Heinrich-Heine University
Institute of Computer Science
Databases and Information Systems
Universitaetsstr.1
D-40225 Duesseldorf, Germany
zhao@cs.uni-duesseldorf.de

ABSTRACT

Data mining is a process of discovering and exploiting hidden patterns from data. Clustering as an important task of data mining divides the observations into groups (clusters), which is according to the principle that the observations in the same cluster are similar, and the ones from different clusters are dissimilar to each other. Subspace clustering enables clustering in subspaces within a data set, which means the clusters could be found not only in the whole space but also in subspaces. The well-known subspace clustering methods have a common problem, the parameters are hard to be decided. To face this issue, a new subspace clustering method based on Bottom-Up method is introduced in this article. It takes a gravitation function to select data and dimensions by using self-comparison technique. The parameter decision is easy, and does not depend on amount of the data, which makes the subspace clustering more practical.

Keywords

data mining, subspace clustering, gravitation, high dimensional data

1. INTRODUCTION

Because of the modern techniques, data collection is nowadays efficient and cost-effective. The data's amount is huge and most data is stored in a raw form, which is not analyzed yet. Usually we need to find out unknown or hidden information from raw data. Data mining is such a process of discovering and exploiting hidden patterns from data. It involves clustering, classification, regression, association, etc.

Clustering divides the observations into groups (clusters), so that the observations in the same cluster are similar, meanwhile, the ones from different clusters are dissimilar. Clustering is important for data analysis, such as market basket analysis, bio science, fraud detection and so on.

Subspace clustering enables clustering in subspaces within a data set, which means that the clusters could be found in subspaces rather than only in the whole space.

1.1 Related works

In a review of subspace clustering [12], the subspace clustering algorithms are categorized into two groups: top-down search and bottom-up search methods. Top-down methods like PROCLUS[1], ORCLUS[2], FINDIT[15], σ-Clusters[16], COSA[5] use multiple iterations for improving the clustering results. By contrast, bottom-up methods find firstly clusters in lower subspaces, and then expand the searching by adding more dimensions. Some examples are CLIQUE [3], ENCLUS [4], MAFIA [6], CBF [10], CLTree [11], DOC [13].

No matter in which group, almost all subspace clustering algorithms have a common problem with finding appropriate values for their parameters. For instance, most of top-down methods have to estimate parameters like number of clusters and subspaces, the clustering results are improved by the iterations that are based on these parameters, which are absolutely not easy to estimate. In bottom-up methods, the key parameters such as density, grid interval, size of clusters, etc. are also hard to be determined. It is necessary to find a method that determines parameters easily, in order to make the clustering job more practical.

DENCLUE [9] is a density-based clustering algorithm by using Gaussian kernel function as its abstract density function and hill climbing method to find cluster centers. DENCLUE 2.0 [8] is an improvement on DENCLUE. The algorithms differ from other density-based approaches in that they calculate density to each data point instead of an area in the attribute space. DENCLUE has not to estimate the number or the position of clusters, because clustering is based on the density information of each point. However it is still necessary to estimate the parameters in these two algorithms, such as σ, ξ in DENCLUE and ϵ, p in DENCLUE 2.0. Besides, they are not designed for subspace clustering.

Applying the Newton's universal law of gravitation in clustering is not a novel idea. A gravitational clustering algorithm [7] simulates the movement of objects by applying the gravitational force, and detects clusters from merged objects. A shrinking-based approach [14] inspired by gravi-

GvD Workshop'10, 25.-28.05.2010, Bad Helmstedt, Germany.

tation is a grid-based clustering method, which shrinks the objects in a grid cell towards the data centroid and finds the clusters. However, for each algorithm we have to find appropriate values for its parameters.

1.2 Contributions of the paper

In this paper, we introduce a new density-based bottom-up subspace clustering method called SUGRA (SUbspace clustering method by using GRAvitation's function). The basic idea is similar to DENCLUE, instead of using the Gaussian kernel function, we use gravitation's function with scaled distance to represent the density function, but the objects don not have to move towards the centroid. From this simple idea we have found an interesting property that in one dimensional subspace almost all cluster objects and non-cluster objects (noise) are separated by a constant. With this property, we can detect clusters very distinctly, meanwhile, SUGRA realizes the reduction of parameters by subspace clustering.

The remainder of this paper is organized as follows. The idea of SUGRA is introduced in section 2, where section 2.1 and 2.2 are definitions about cluster and gravitation function respectively, section 2.3 presents the algorithm of SUGRA. The last section contains conclusions and areas of future work.

2. SUBSPACE CLUSTERING WITH GRAVITATION

2.1 Definition of data set and subspace cluster

A data set consists of objects and their attributes. Usually, all objects have common attributes in a data set, such as color, price, length etc., and every object has a value for an attribute. The values that are related to an attribute are in the same domain and conform to the same constraints. A data set could be described as a table, where the objects are just rows, meanwhile the attributes are columns. The attributes could also be considered as dimensions, so that each attribute represents one dimension, and then the objects are points in these dimensions.

In order to describe the attributes and objects clearly, they are defined as follows:

Definition 1. (Data set) Generally, a data set $\mathcal{D}$ could be considered as a pair, which is the combination of $\mathcal{A}$ and $\mathcal{O}$:

$$\mathcal{D} = (\mathcal{A}, \mathcal{O}) \tag{1}$$

where $\mathcal{A}$ is a set of all attributes (dimensions), and $\mathcal{O}$ is a set of all objects:

$$\mathcal{A} = \{a_1, a_2, \cdots, a_i, \cdots\}, \quad \mathcal{O} = \{o_1, o_2, \cdots, o_p, \cdots\} \tag{2}$$

where o_p is an object with values on $\mathcal{A}$:

$$o_p = \{o_p^{a_1}, o_p^{a_2}, \cdots, o_p^{a_i}, \cdots\} \tag{3}$$

We denote the values of all objects on attribute a_i with:

$$o^{a_i} = \{o_1^{a_i}, o_2^{a_i}, \cdots, o_p^{a_i}, \cdots\} \tag{4}$$

Definition 2. (Subspace cluster) A subspace cluster S is also a data set and defined as follows:

$$S = \widetilde{\mathcal{D}} = (\widetilde{\mathcal{A}}, \widetilde{\mathcal{O}}) \tag{5}$$

where $\widetilde{\mathcal{A}} \subseteq \mathcal{A}$ and $\widetilde{\mathcal{O}} \subseteq \mathcal{O}$, and S must satisfy a particular condition $\mathcal{C}$, which is defined differently in every subspace clustering algorithm.

A subspace cluster S could then be written like this:

$$S = (\widetilde{\mathcal{A}}, \widetilde{\mathcal{O}}) = (\{a_1, a_{12}, a_{60}, \cdots\}, \{o_1, o_5, o_{30}, \cdots\})$$

The cardinality regarding the objects and the dimensions in S are defined respectively:

$$|S|_O = |\widetilde{\mathcal{O}}|, \quad |S|_A = |\widetilde{\mathcal{A}}| \tag{6}$$

REMARK 1. *Suppose S_1, S_2 are two subspace clusters, where $S_1 = (\mathcal{A}_1, \mathcal{O}_1)$ and $S_2 = (\mathcal{A}_2, \mathcal{O}_2)$, then*

- *If $\mathcal{A}_1 \neq \mathcal{A}_2 \vee \mathcal{O}_1 \neq \mathcal{O}_2 \Longrightarrow S_1 \neq S_2$, the subspace clusters with different dimensions or objects are considered as different ones.*
- *$\forall \mathcal{A}' \subseteq \mathcal{A}_1$, $S' = (\mathcal{A}', \mathcal{O}_1)$ is also a subspace cluster.*
- *If $\mathcal{A}_1 \subseteq \mathcal{A}_2 \wedge \mathcal{O}_1 = \mathcal{O}_2$ or $\mathcal{A}_1 = \mathcal{A}_2 \wedge \mathcal{O}_1 \subseteq \mathcal{O}_2 \Longrightarrow S_1 < S_2$. Only the largest subspace cluster will be taken in the clustering result.*

Definition 3. The intersection of two subspace clusters is defined as follows:

$$S_1 \cap S_2 = (\mathcal{A}_1 \cup \mathcal{A}_2, \mathcal{O}_1 \cap \mathcal{O}_2) \tag{7}$$

Definition 4. $\mathcal{S}^{a_i}$ is the set of all subspace clusters found in dimension a_i, $\mathcal{S}^{\mathcal{D}}$ is the set of all subspace clusters found in $\mathcal{D}$, finding $\mathcal{S}^{\mathcal{D}}$ is the task of subspace clustering.

2.2 Gravitation

Gravitation is a natural phenomenon, which describes the force of attraction between objects with mass. The gravitation is important, because it influences our normal lives.

The Newton's law of universal gravitation is defined as follows:

$$G = \mathcal{G} \cdot \frac{m_1 m_2}{r^2} \tag{8}$$

where G is the gravity between two point masses, $\mathcal{G}$ is the gravitational constant, m_1 and m_2 are the masses of two points respectively, r is the distance between the two point masses.

SUGRA tries to use the gravitation's function for the measurement between the objects. In order to make the calculation easier, a simple gravity function is used here:

Definition 5. (Simple gravity function)

$$G_{pq}^{a_i} = \frac{m_p m_q}{r_{pq}^2} \tag{9}$$

Suppose that a single object o_p has a mass $m_p = 1$, r_{pq} is the distance between o_p and o_q is defined as $r_{pq} = \dfrac{l_{pq}}{L/(N-1)}$,

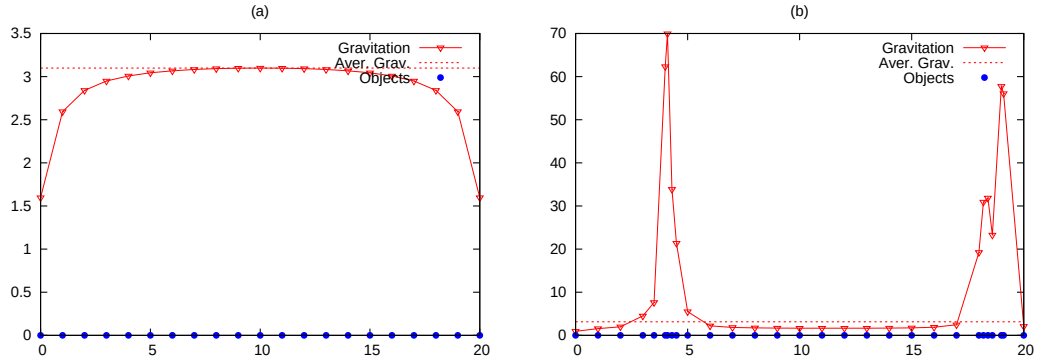

Figure 1: Properties of gravitation in one dimension

where $L = max(o^{a_i}) - min(o^{a_i})$ is the length of this dimension and $N = |\mathcal{O}|$ is the number of the objects. r_{pq} indicates a proportion of the real length l_{pq} to the average interval $L/(N-1)$, so that

$$G_{pq}^{a_i} = \frac{1}{\left(\frac{l_{pq}}{L/(N-1)}\right)^2} = \frac{L^2}{l_{pq}^2(N-1)^2} \tag{10}$$

REMARK 2. *If $r_{pq} = 0$, o_p and o_q stand at a same place in a_i. In order to let $G_{pq}^{a_i}$ calculable and to get a logical result, we should set r_{pq} greater than 0 but smaller than any other distances. An idea is setting r_{pq} to a half of the minimum distance in a_i. For example, o_m, o_n has the minimum distance $r_{mn} > 0$ in a_i, then setting $r_{pq} = r_{mn}/2$ make sure that $G_{pq}^{a_i} > G_{mn}^{a_i}$, which is expected.*

REMARK 3. *The r_{pq} is such defined as a proportion distance but not a real distance because that it enables the data with different ranges of values to be calculated into a same range. For example, the attributes* `age` *and* `salary` *are obviously in two ranges, but by using such a proportion distance the two attributes could be calculated and compared together.*

There are further definitions, which are important for the SUGRA algorithm.

Definition 6. (Gravitation of an object) The gravitation of an object o_p in dimension a_i is defined as the sum of gravitation from o_p with other objects.

$$G_p^{a_i} = \sum_{\forall q,\ q \neq p} G_{pq}^{a_i} \tag{11}$$

REMARK 4. *The gravitation of an object defined in (11) has following properties in one dimension:*

- *An object in the middle has a greater value of gravitation than one at the edge, which could be clearly seen, if the objects are distributed uniformly (see Figure 1 (a)).*
- *An object that lies near to others (cluster objects) has a greater gravitation than that of objects far from others (non-cluster objects) (see Figure 1 (b)).*

Definition 7. (Average gravitation) The average gravitation $\overline{G^{a_i}}$ of dimension a_i is the gravitation of the middle object of uniformly distributed objects in a_i. $\overline{G^{a_i}}$ is presented with a dotted line in *Figure 1*.

Suppose o_m is the object in the middle. $\overline{G^{a_i}}$ could be calculated as follows:

$$\begin{aligned}
\overline{G^{a_i}} &= \sum_{\forall p,\ p \neq m} \frac{L^2}{l_{mp}^2(N-1)^2} = \frac{L^2}{(N-1)^2} \cdot \left(\sum_{\forall p,\ p \neq m} \frac{1}{l_{mp}^2} \right) \\
&= \frac{L^2}{(N-1)^2} \cdot \left(\sum_{1 \leq n \leq \frac{N}{2}} \frac{1}{(\frac{L}{N-1} \cdot n)^2} \right) \cdot 2 \\
&= 2 \cdot \left(\sum_{1 \leq n \leq \frac{N}{2}} \frac{1}{n^2} \right) \underset{N \to \infty}{\longrightarrow} 2 \cdot \frac{\pi^2}{6} = \frac{\pi^2}{3} \approx 3.29
\end{aligned}$$

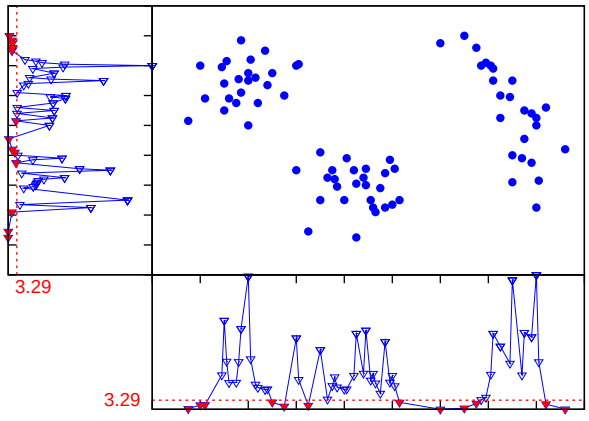

Figure 2: SUGRA on two dimensional data

REMARK 5. *The gravitation values of cluster objects and*

non-cluster objects have great differences. The non-cluster objects have usually very small values of gravitation, meanwhile the cluster objects have larger values. This property is very important for the clustering process.

If a data set has many objects, which means N is a big number, then $\overline{G^{a_i}} \approx 3.29$*, from experiments we found out that using the average gravitation* $\overline{G^{a_i}}$ *as a threshold to separate cluster and non-cluster objects returns good results. This is not a silver bullet, but can be thought as a starting point, the threshold could be regulated near this value.*

Figure 2 represents an example of SUGRA on two dimensional data. The value 3.29 have separated the gravitation on the two dimensional space respectively, where the red points illustrates the gravitation of rand points.

2.3 Algorithm of SUGRA

This algorithm consists of two steps:

1. Data selection (Clustering in one dimensional spaces)
2. Dimension selection (Clustering in high dimensional spaces)

As a Bottom-Up algorithm, SUGRA handles data firstly in one dimensional space, because one dimensional data can be dealt with easily. Finding clusters in high dimensional space is based on the clusters found in one dimension.

2.3.1 Data selection

Algorithm 1: Data selection

Input: $\mathcal{D} = (\mathcal{A}, \mathcal{O})$
Output: $\mathcal{S}^{a_i}$

```
foreach a_i ∈ A do
    Sort o^{a_i}
    initialize t=1
    foreach o_p^{a_i} ∈ o^{a_i} do
        if G_p^{a_i} > Ḡ^{a_i} then
            if o_{p-1}^{a_i} ∈ S_t^{a_i} and |o_p^{a_i} − o_{p-1}^{a_i}| < L̄ then
                add o_p^{a_i} to S_t^{a_i}
            else
                t++
                add o_p^{a_i} to S_t^{a_i}
            end
        end
    end
end
```

As discussed above, the clusters are firstly selected on each dimension through the gravitation. First of all, o^{a_i} are sorted in ascending order. For example, if $G_p^{a_i}$ has a greater value than the threshold $\overline{G^{a_i}}$, $o_p^{a_i}$ is then chosen as a cluster-candidate. If its neighbor $o_{p-1}^{a_i}$ is also a cluster-candidate and their distance is smaller than the average distance, they will be considered in one cluster, otherwise $o_p^{a_i}$ is set into a new cluster. The process will stop when there is no more new cluster found in a_i. The processes are the same for other dimensions. *Algorithm 1* shows more details about the data selection.

After the data selection process, we get subspace clustering results in every one dimensional space $\mathcal{S}^{a_i}$, a subspace cluster $S_t \in \mathcal{S}^{a_i}$ could have the form like

$$S_t = (\{a_i\}, \{o_1, o_5, o_9, \cdots\}) \tag{12}$$

2.3.2 Dimension selection

Algorithm 2: Dimension selection

Input: S^{a_i}
Output: $\mathcal{S}^{\mathcal{D}}$

```
add all S_t ∈ S^{a_i} to S^D
foreach S ∈ S^D do
    while find S' ∈ S^D and S ≠ S' do
        if |S ∩ S'|_O ≥ 2 then
            add S ∩ S' to S^D
        end
    end
end
return
```

In data-selection, the one dimensional clusters were found with the forms like (12), the finding of subspace cluster in high dimension is just based on the intersection defined in (7). For subspace clusters S_1 and S_2, if $|S_1 \cap S_2|_O \geq 2$, then $S_1 \cap S_2$ is a new subspace cluster.

Every combination of clusters should be checked through the intersection, this process will stop when no more new cluster is found. The final result will keep only the largest subspace clusters. The detailed algorithm is shown in *Algorithm 2*.

2.4 Further discussions

The choosing of parameters is usually difficult for a subspace clustering algorithm, a little deviation may cause a different result. The boundaries between cluster objects and non-cluster objects are especially indistinct and they could be recognized hardly. SUGRA uses the gravitation function that marks the cluster and non-cluster objects with great differences, which makes the parameter decision easily.

Data selection. The experimental experience shows that $\overline{G^{a_i}}$ could separate the cluster objects and non-cluster objects very well by data selection, as defined in *Definition 7*, $\overline{G^{a_i}} \approx 3.29$ does not depend on $|\mathcal{O}|$ and could be used as threshold for almost all situations. If the results are not satisfying, the threshold could be set a little smaller or greater.

Dimension selection. The condition $|S_1 \cap S_2|_O \geq 2$ is used in dimension selection to decide, whether $S_1 \cap S_2$ is a subspace cluster. The condition indicates that an object-group with more than two objects will be taken as a new cluster. This setting has a high precision, because not only big clusters but also small clusters could be found, but naturally it takes much time. In contrast, choosing a greater number may gain time but lose some interesting small clusters.

2.4.1 Run time

The run time of the data selection in a dimension a_i is $|\mathcal{O}|^2$, and for $\mathcal{D} = (\mathcal{A}, \mathcal{O})$ is $|\mathcal{A}| \cdot |\mathcal{O}|^2$.

In dimension selection, every possible combination of subspace clusters could be examined, so the maximum run time of dimension selection is 2^m, where m is the number of one dimensional subspace clusters found in data selection.

3. CONCLUSIONS

Subspace clustering is able to discover clusters and extract their features from the subspace of high dimensional data, which is commonly gathered in many fields. Most familiar subspace clustering approaches have the problems with determining the parameters. We attempt to apply the gravitation function in subspace clustering in order to find out a new method make the determination of the parameters easier. The method is named SUGRA, which belongs to Bottom-Up algorithms. Firstly, it finds out clusters by using gravitation function in one dimensional space, then it combines the clusters in higher dimensions for searching high dimensional subspace clusters.

In SUGRA, the non-cluster objects have always low gravitation values (<3.29), meanwhile the cluster objects have very large values, which depend on the clusters' density and number of objects. The value 3.29 does not depend on the number of objects, so it enables separating the non-cluster objects in order to get cluster objects. We don't have to choose parameters like other algorithms, SUGRA can get almost a satisfying result for a start by using this threshold.

The future research will be focused on optimizing the gravitation function and the algorithm in order to improve the subspace clustering results. The gravitation technique should be used not only in one dimensional data but also directly in multiple dimensions. Another work is to let SUGRA adapt various data types, such as categorical data.

4. REFERENCES

[1] C. C. Aggarwal, J. L. Wolf, P. S. Yu, C. Procopiuc, and J. S. Park. Fast algorithms for projected clustering. In *Proceedings of the 1999 ACM SIGMOD international conference on Management of data*, pages 61–72, Philadelphia, Pennsylvania, United States, May 31-June 03 1999.

[2] C. C. Aggarwal and P. S. Yu. Finding generalized projected clusters in high dimensional spaces. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pages 70–81, Dallas, Texas, United States, May 15-18 2000.

[3] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. In *Proceedings of the 1998 ACM SIGMOD international conference on Management of data*, pages 94–105, Seattle, Washington, United States, June 01-04 1998.

[4] C.-H. Cheng, A. W. Fu, and Y. Zhang. Entropy-based subspace clustering for mining numerical data. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 84–93, San Diego, California, United States, August 15-18 1999.

[5] J. H. Friedman and J. J. Meulman. Clustering objects on subsets of attributes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2002.

[6] S. Goil, H. Nagesh, and A. Choudhary. Mafia: Efficient and scalable subspace clustering for very large data sets. Technical Report CPDC-TR-9906-010, Northwestern University, June 1999.

[7] J. Gomez, D. Dasgupta, and O. Nasraoui. A new gravitational clustering algorithm. In *In Proc. of the SIAM Int. Conf. on Data Mining (SDM*, 2003.

[8] A. Hinneburg and H.-H. Gabriel. Denclue 2.0: Fast clustering based on kernel density estimation. In *Proc. of International Symposium on Intelligent Data Analysis 2007 (IDA'07)*, Ljubljana, Slowenien, 2007. LNAI Springer.

[9] A. Hinneburg and D. A. Keim. An efficient approach to clustering in multimedia databases with noise. In *Proc. 4rd Int. Conf. on Knowledge Discovery and Data Mining*, New York, 1998. AAAI Press.

[10] D.-S. J. Jae-Woo Chang. A new cell-based clustering method for large, high-dimensional data in data mining applications. In *Proceedings of the 2002 ACM symposium on Applied computing*, pages 11–14, Madrid, Spain, March 2002.

[11] B. Liu, Y. Xia, and P. S. Yu. Clustering through decision tree construction. In *Proceedings of the ninth international conference on Information and knowledge management*, pages 20–29, McLean, Virginia, United States, November 06-11 2000.

[12] L. Parsons, E. Haque, and H. Liu. Subspace clustering for high dimensional data: A review. *Sigkdd Explorations*, 6:90–105, June 2004.

[13] C. M. Procopiuc, M. Jones, P. K. Agarwal, and T. M. Murali. A monte carlo algorithm for fast projective clustering. In *Proceedings of the 2002 ACM SIGMOD international conference on Management of data*, Madison, Wisconsin, June 03-06 2002.

[14] Y. Shi, Y. Song, and A. Zhang. A shrinking-based approach for multi-dimensional data analysis. In *VLDB '2003: Proceedings of the 29th international conference on Very large data bases*, pages 440–451. VLDB Endowment, 2003.

[15] K.-G. Woo and J.-H. Lee. *FINDIT: a Fast and Intelligent Subspace Clustering Algorithm using Dimension Voting*. PhD thesis, Korea Advanced Institute of Science and Technology, Taejon, Korea, 2002.

[16] J. Yang, W. Wang, H. Wang, and P. Yu. δ-clusters: Capturing subspace correlation in a large data set. In *Proceedings of the 18th International Conference on Data Engineering*, page 517, February 26-March 01 2002.

[17] J. Zhao. Automatische Parameterbestimmung durch Gravitation in Subspace Clustering. *21. Workshop "Grundlagen von Datenbanken", Rostock*, 2009.