Efficient Monitoring of Changing Clusters on Multi-dimensional Data Streams

Nam Hun Park[1], Kil Hong Joo[2*] and Su Young Han[1]

[1]Dept. of Computer Science, Anyang University, 102 Samsungli, Buleunmyun, Ganghwagun, Incheon, Korea, 417-833
{nmhnpark, syhan}@anyang.ac.kr

[2]Dept. of Computer Education, Gyeongin National University of Education, San 6-8 Seoksudong Manangu Anyangsi, Gyeonggi, Korea, 430-040
khjoo@ginue.ac.kr
*Corresponding Author

Abstract. A real-life data stream usually contains many dimensions and some dimensional values of its data elements may be missing. In order to effectively extract the on-going change of a data stream with respect to all the subsets of the dimensions of the data stream, the abilities to trace its subspace clusters and to predict the change are very important. In this paper, the index structure is adopted for subspace clustering over a data stream. To cluster adaptively with the change of data streams, the supports of ranges are monitored and incoming change of supports are predicted.

Keywords: Data Streams, Clustering, Data mining, Grid-based clustering, Adaptive memory utilization

1 Introduction

Data mining researches on data streams are motivated by emerging applications involving continuous massive data sets such as customer click streams, multimedia data and sensor data. A real-life data stream usually contains many dimensions and some dimensional values of its data elements may be missing[1]. In order to effectively extract the on-going change of a data stream with respect to all the subsets of the dimensions of the data stream, the abilities to trace its clusters and to predict the change are very important[2,3].

In this paper, the grid-based index structure is adopted for subspace clustering over a data stream. Given a predefined sequence of dimensions $N_1 \rightarrow N_2 \rightarrow \ldots \rightarrow N_n$, initially an independent grid-list for each dimension monitors its one-dimensional clusters at the first level of a monitoring tree. When a grid-cell of the grid-list for the dimension $N_k (1 \leq k \leq n)$ becomes a dense unit grid-cell, a set of new grid-lists are created as the children of the grid-cell. In order to enumerate all the possible two-dimensional subspaces of the dimension N_k uniquely, only for those dimensions which are after the dimension N_k in the dimension sequence, new grid-cell lists are created. Consequently,

there are *(d-k)* distinct grid-cell lists are created as the children of the grid-cell. A grid cell of a node in the k^{th} level of a monitoring tree is corresponding to a rectangular subspace formed by intersecting the intervals of the grid-cells in the path from the root to the node containing itself. Also, to reflect the change of data streams in a real-time, the support of a grid-cell is monitored and predicted by measuring velocity of density change.

2 Monitoring Data Streams

Given a data stream of an *n*-dimensional data space $N=N1\mathbf{x}...\mathbf{x}Nn$, the region of a *k*-dimensional grid-cell ($1 \le k \le n$) can be defined by a set of *k* intervals each of which lies in a distinct dimension. The rectangular space of a *k*-dimensional grid-cell defined by dimensions N_1, N_2,...,N_k is $RS=I1\mathbf{x}I2\mathbf{x}...\mathbf{x}Ik$ where I_1 , I_2,..., and I_k are intervals in the dimension N_1, N_2,..., N_k respectively. To monitor the distribution statistics of data elements in the rectangular space of such a grid-cell efficiently, a monitoring tree defined in Fig. 1. is employed.

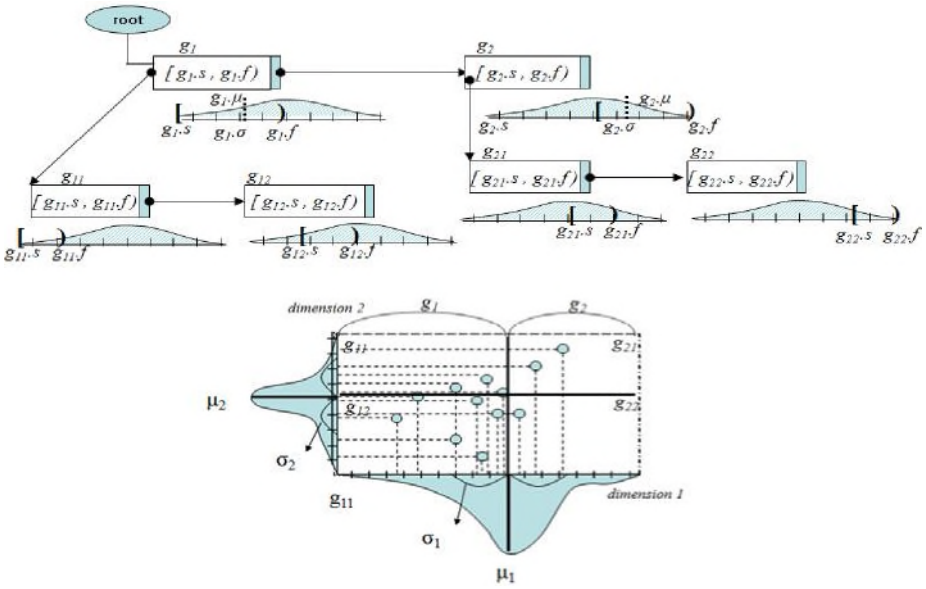


Fig. 1. An example of grid-cells and data space

Given a predefined sequence of dimensions N_1—'...—'N_n and a partitioning factor *h*, grid-cell lists $\boldsymbol{L_1,..., L_n}$ are created to maintain the one-dimensional grid-cells of each one-dimensional data space respectively. Initially, each grid-cell list at the first level maintains *h* initial grid-cells and a single node is created to form each grid-cell list. As a new object o^t is arrived, the previous statistics are update as follows:

For the continuously generated data elements of a data stream, dense grid-cells in each grid-cell list $\boldsymbol{L_v}(1 \le v \le n)$ are recursively partitioned into *h* smaller grid-cells. Each

of the child nodes is the head node of a new grid-cell list for two-dimensional grid-cells. The grid-cell list created for the dimension N_l ($v+1 \le l \le n$) monitors the on-going distribution statistics of data elements in the two-dimensional rectangular subspace space $_{gp1.I \times Nl.}$ Given a grid-cell $g_q^2(I,c,\mu,\sigma)$ of the new grid-cell list in the second level, the two-dimensional rectangular subspace denoted by the grid-cell $g_q^2(I,c,\mu,\sigma)$ is $g_p^1.I \times g_q^2.I$.
When a grid-cell in the grid-cell list for the dimension N_l ($v+1 \le l \le n$) in the second level becomes dense, it is also partitioned into smaller grid-cells. Consequently, the number of grid-cells in the grid-cell list is increased. Furthermore, when it becomes a dense unit grid-cell, *(n-l)* new grid-cell lists for the subsequent dimensions are created in the third level as the children of the dense grid-cell as well.

3 Prediction of the grid-cell support

For a grid-cell in a monitoring tree, its support per time changes over time. However, analyzing the past frequency of a grid-cell can help to predict its support in the future. However, to predict the support more accurately over time, more information should be maintained each grid-cell entry.
For a grid-cell, its support velocity is defined as the difference of its support. The V_{count} of the grid-cell means its velocity in the most recent time. When the current time is t^{th} time, the support of $(t+1)^{th}$ time can be predicted from V_{count}.

$$V_{count^{t+1}} = count^t - count^{t-1} \quad (1)$$

From the velocity V_{count} $_{t+1.}$ the count at $(t+1)^{th}$ time can be predicted as follows:

$$P_{count^{t+1}} = count^t + V_{count}^{t+1} \quad (2)$$

The support at $(t+1)^{th}$ time can be predicted from $_{Pcountt+1}$ as follows:

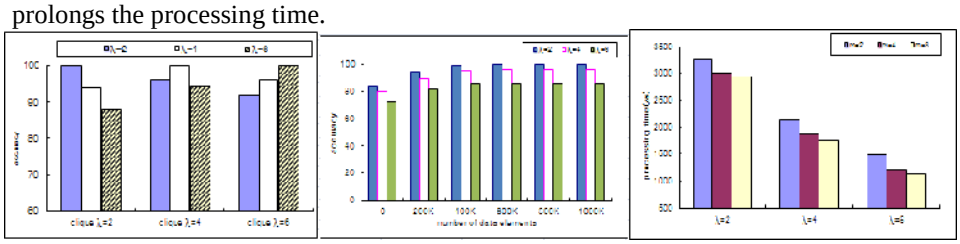
$$S^{t+1} = (count^t + {}_{Pcountt+1}) / (|D^t| + count^{t+1})$$

Let a grid-cell with the current support S^t would become a cluster after v times. Then, S^{t+v}=($count^t$ + $_{Pcountt+v}$) / ($|D^t|$ + $count^{t+v}$) ≥ S_{min} is satisfied. From equations (1) and (2), the support is predicted by solving the time v.

4 Experiments

In order to analyze the performance of the proposed method, a data set containing one million 20-dimensional data elements is generated by the data generator used. The accuracy of the proposed method is presented in Figure 2. CLIQUE[4] is a well-known conventional grid-based subspace clustering algorithm for a finite data set and it is used as a yardstick to measure the accuracy of the proposed method. Figure 2-(a) illustrates the accuracy of the proposed method for the four different values of X. When the value of X for the proposed method is the same as that for CLIQUE, these two methods have the same accuracy. Figure 2-(b) shows the variation of the accuracy as new data elements are generated. Since lots of partitioning operations are

occurred to find unit grid-cells in the early stage of subspace clustering, the accuracy of the proposed method is relatively low. However, as unit grid-cells are found by consecutive partitioning operations, the accuracy is increased gradually. Figure 2-(c) shows the processing time of the proposed method. When the order is too small, i.e. m=2, the number of sibling entries in each sibling list is increased rapidly, which prolongs the processing time.



(a) Accuracy comparison (b) Accuracy variation (c) Memory usage

Fig. 2. Performance evaluations

5 Conclusion

As the number of dimensions for a data set is increased, subspace clustering is useful to analysis interesting groups in the subsets of the dimensions. However, because conventional subspace clustering methods need to create all the possible candidate clusters and examine the data elements of a data set repeatedly for each candidate. They can not be used for an on-line data stream. In this paper, we have proposed a subspace clustering method over a data stream. By maintaining grid-based structure, the current statistics of a data stream are carefully monitored. As the support of each grid-cell is predicted with the support velocity, the rapid change of a data stream can be predicted for the real-time data mining.

References

1. Ming Hua, Jian Pei and Xuemin Lin. Ranking queries on uncertain data. The International Journal on Very Large Data Bases. Vol. 20, No. 1, pages 129-153, February, 2011.
2. Liadan O'Callaghan, Nina Mishra, Adam Meyerson, Sudipto Guha, and Rajeev Motwani. STREAM-data algorithms for high-quality clustering. In Proc. of IEEE International Conference on Data Engineering, March 2002
3. Charu C. Aggarwal, Jiawei Han, Jianyong Wang, Philip S. Yu. A Framework for Clustering Evolving Data Streams. In Proc. VLDB 29th, Berlin, 2003
4. Hans-Peter Kriegel, Peer Kroger, Matthias Renz and Sebastian Wurst. A Generic Framework for Efficient Subspace Clustering of High-Dimensional Data. In Proceedings of the Fifth IEEE International Conference on Data Mining, pages 250-257, 2005