Enhanced clustering

Gianluca Bortoli
DISI - University of Trento
Student id: 179816
gianluca.bortoli@studenti.unitn.it

Martin Brugnara
DISI - University of Trento
Student id: 182904
martin.brugnara@unitn.it

ABSTRACT

Keywords

Big data, Data mining, Clustering, Streaming, Parallel computation

1. INTRODUCTION

The clustering problem received a lot of attention in the data mining [3], statistics [5, 1] and machine learning literatures. Furthermore it has been exploited in many other disciplines.

Generally speaking, the problem consists in grouping together data items that are "similar" to each other. The concept of similarity varies a lot in the different contexts it can be applied. For example the Euclidean distance (L2) can be used when dealing with continuous values or the Jaccard similarity index, which computes similarity for generic sets of elements. Nonetheless, the underlying algorithm is agnostic with respect to the similarity measure that is applied to compute a distance between the elements in the data. Clustering can be also viewed as identifying the dense regions of the probability density of the data source [2].

The literature suggests two different approaches: *iterative* and *hierarchical* algorithms.

The first strategy usually needs some parameters to be set and known in advance. For example the *k-means*, which is one of the most popular and adopted algorithm, requires the number of cluster to be found (K).

The latter can be implemented both in a top down (divisive) or a bottom up (agglomerative) manner. Initially, the divisive algorithm treats all data as a single big cluster and later splits it until every object is separated [4]. On the contrary, the agglomerative starts considering each "element" as a *singleton* (a cluster composed of one element). Next, the most similar clusters are collapsed together until only one big cluster remains. Implicitly the merging order defines a clear hierarchy among the intermediate representations (dendrogram).

Clearly, both the above mentioned approaches to the clustering problem have their disadvantages. The iterative methods require prior knowledge on the data distribution, while the hierarchical ones imply the user interaction to decide the dendrogram's cut height. A solution that does not suffer from those is still an open challenge.

In this work we propose a completely autonomous system which merges the two strategies to overcome their weaknesses, meaning that it satisfies the following *Data Mining Desiderata*:

1. **streaming**: require one scan of the database, since reading from secondary memory is still the most costly I/O operation. Moreover the analysis can be stopped and restarted without having to re-process the whole data ("stop and resume" support). This property adds the capability to incorporate additional data with existing model efficiently (incremental computation).
2. **on-line "anytime" behaviour**: a "best" answer is always available at any time during the computation phase.
3. **limited memory**: the tool must work within the bounds of a given amount of main memory (RAM).

2. RELATED WORK

3. PROBLEM DEFINITION

4. SOLUTION

5. CONCLUSIONS AND FUTURE WORK

6. REFERENCES

[1] J. D. Banfield and A. E. Raftery. Model-based gaussian and non-gaussian clustering. *Biometrics*, pages 803–821, 1993.

[2] P. S. Bradley, U. M. Fayyad, C. Reina, et al. Scaling clustering algorithms to large databases. In *KDD*, pages 9–15, 1998.

[3] U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy. *Advances in knowledge discovery and data mining*, volume 21. AAAI press Menlo Park, 1996.

[4] L. Kaufman and P. J. Rousseeuw. *Finding groups in data: an introduction to cluster analysis*, volume 344. John Wiley & Sons, 2009.

[5] H. C. Tijms. *Stochastic models: an algorithmic approach*, volume 303. John Wiley & Sons Inc, 1994.