Письма в ЭЧАЯ. 2015. Т. 12, № 1(192). С. 237–240

WATERSHED ON VECTOR QUANTIZATION FOR CLUSTERING OF BIG DATA

S. V. Mitsyn [1], *G. A. Ososkov* [2]

Joint Institute for Nuclear Research, Dubna

A method for clustering large amounts of data is presented which is a sequenced composition of two algorithms: the former builds a partition of input space into Voronoi regions and the latter clusters them. First, a model of clusters as high-density regions in input space is presented, then it is shown how a Voronoi partition and its topological map a) can be built and b) used as a low complexity approximation of the input space. During the b) step, the usage of "watershed" algorithm is presented which has been previously used for image segmentation, but it is its application to a data space partition that is proposed by the authors.

В данной статье представлен метод кластеризации данных большого объема в виде последовательной композиции двух алгоритмов: первый строит разбиение входного пространства на области Вороного, а второй кластеризует их. Во-первых, представлена модель кластеризации данных как областей большой плотности во входном пространстве, затем показано, как разбиение Вороного и его топология могут быть а) построены и б) использованы как упрощенное приближение входного пространства. В течение шага б) показано действие алгоритма «Водораздел», который часто используется для сегментации изображений, но это его первое применение в разбиении пространства входных данных, известное авторам.

PACS: 07.05.Hd; 07.05.Pj; 07.05.Rm

INTRODUCTION

Data clustering is a data mining method, during which objects comprising some data set are partitioned to a set of disjoint sets, whose union contains all initial objects. The goal of clustering is to give non-strict, heuristic, simplified description of a data set by splitting it into several groups according to a closeness of objects in the feature space for each of groups. As a final result, some idea about structure of the data is obtained.

Big data, when applied to data mining, represents a fact that analyst is facing a too large data set so that is hard to deal with on common computers. Particular characteristics of such a situation include a large number of objects and measurements (starting from 10^6) and dimensions. Standalone difficulty is an absence of apriori hypothesis about data structure — number of clusters, their form and position. Our task is to develop a method for big data clustering.

[1]E-mail: svm@jinr.ru
[2]E-mail: ososkov@jinr.ru

MODEL

For each object of data set let us assign a vector in multidimensional vector space (feature space), whose components describe quantitatively characteristics of corresponding object. The task of clustering is to split objects into groups by their closeness in vector space.

First, we need to define what the nature of sought-for groups is and how they influence the population (overall) distribution. We take up the simple variant, where every cluster is a "dense" group of objects, distributed with a single well-defined maximum of distribution density in a feature space.

Let $p(x)$ be a density distribution function in a population with x being a vector of a feature space. Recall that analytical form of $p(x)$ is unknown, but a big sample is given. Our goal is to find partition $c(x) \in \mathbb{N}$, $c(x_k) = c(x_l) \Leftrightarrow$ objects k and l belong to one group. We'll say that a group can be identified if a) a peak (local maxima) in $p(x)$ can be set to correspondence to this group and b) a cluster can be formed around the peak to distinguish the group from other groups.

ALGORITHM

The algorithm is a multistep clustering, where on each step a qualitatively new data description is discovered.

On the first step, the initial data partition is built — Voronoi partition, with sets being convex and contained inside (hyper-)polyhedron, so that they unite objects which are close to each other. It can be formalized as a mapping $v(x) : X \to V$, $V \subset \mathbb{N}$, which is a primary clustering. The next steps give more precise result by clusters merging.

On the second step, a similarity relation is built on Voronoi sets — that is, a set of unordered pairs $(i, j) \in E$. Having a set of Voronoi sets and a set of pairs, we can define a graph $G = \langle V, E \rangle$ with V being a set of vertices (set of indices corresponding to Voronoi sets) and E being a set of edges defined previously. For each edge a weight $w_{i,j}$ is assigned, which increases monotonically with distance between clusters i and j (which means that clusters are less similar).

On the third step, the set of Voronoi regions is partitioned by G graph cut with Minimal Spanning Tree relative to minima method, which is the final result.

Voronoi Regions. A partitioning of Euclidean space is defined by a set of points — a codebook, and points are called Voronoi region centers. Voronoi cell is a region of vector space, where all points are closer (by Euclidean distance) to its center rather than to the center of any other Voronoi cell.

There exist many algorithms for Voronoi partitioning, e.g., K-means [2], and algorithms that are based on it — fuzzy K-means [3], neural methods (Kohonen Self-Organizing Maps [4], Neural Gas, Growing Neural Gas [5]). The listed algorithms are able to handle with sample as well, not only with an explicit density function. Stochastic ones have complexity, independent of sample size, and thus can be used with big data samples.

Similarity Relation. For similarity relation on Voronoi regions one can use at least two strategies: Delaunay triangulation [6] and complete graph. In the former case, the relation is built, where a pair (i, j) is included in E iff polyhedra i and j have common faces. The important implication is that all edges of Minimal Spanning Forest relative to minima [7] of

complete graph are included in E, while skipping vast amount of edges, which may reduce complexity on the third step of our multistep algorithm without changing the result.

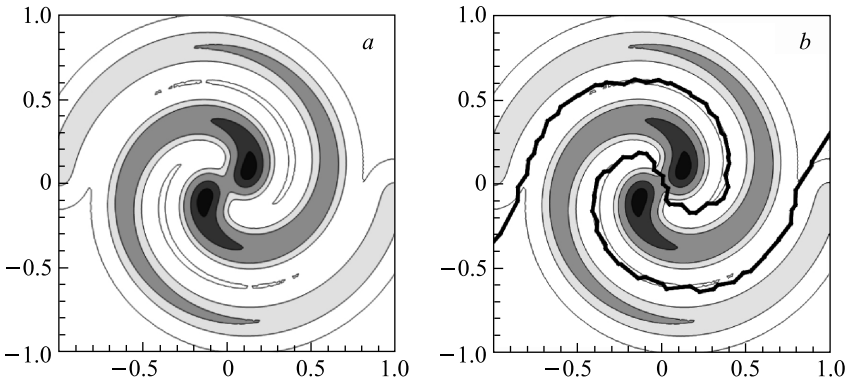
The Watershed. The algorithms listed on the previous step build such a Voronoi partitioning that regions volume $1/q(x)$ tend to decrease with increasing distribution density $p(x)$: $q(x) = C\left[p(x)^{\frac{n}{n+2}}\right]$, with n being space dimensionality. The distance between region centers also decreases, thus weights $w_{i,j}$ of the edges connecting these regions decrease too. This heuristics allows us to propose a hypothesis that $w_{i,j}$ are lower at the maxima of $p(x)$ and higher among periphery.

This in turn allows us to reduce the problem to a graph cut, where each component includes exactly one edge with locally minimal weight, and edges with high weight are removed. This is equivalent to Minimal Spanning Forest relative to minima, which is well studied in, e.g., [7] as an edge watershed method.

The Final Notes on the Watershed. As a watershed result, some false clusters are usually identified as small perturbations in edge distances which occur due to stochastic nature of sample and utilized quantization algorithm. Thus a fourth step, post-merging, should be done. One of possibilities is usage of border dynamics [8], in which each cluster (which is called basin) is assigned a measure, describing its significance, as well as an order relation which defines to which cluster this one has to be merged if it is too insignificant.

EXAMPLE OF SYNTHESIZED DATA CLUSTERING

Particularly interesting clustering cases that are traditionally hard to clusterize are classes that have non-convex twisted form in a feature space. Here we present two spiral classes in the figure, *a*. It is interesting to note that these spirals are themselves non-uniform — distribution density between their heads is higher than density along their tails. Such a case would be hard to deal with using, e.g., single linkage as in [9]. By our algorithm these spirals are separated successfully as in the figure, *b*. This is possible due to high localization of data that watershed uses. Thus, relatively high distribution density between heads does not prevent clustering due to the lack of opaqueness on the tails.



Clustering example for two spirals: *a*) distribution density in feature space; *b*) final segmentation

CONCLUSIONS

A method for Euclidean space clustering is presented that is able to handle big data through its sequential simplification. As a result, analyst that works with data can acquire more qualitative information about the type and structure of data distribution.

REFERENCES

1. *Duran B. S., Odell P. L.* Cluster Analysis: A Survey. Berlin; New York: Springer-Verlag, 1974. 137 p.
2. *MacQueen J.* Some Methods for Classification and Analysis of Multivariate Observations // Proc. of the Fifth Berkeley Symp. on Mathematical Statistics and Probability / Eds.: Le Cam L. M., Neyman J. V. 1. Berkeley: Univ. California Press, 1967. P. 281–297.
3. *Bezdek J. C.* Pattern Recognition with Fuzzy Objective Function Algorithms. 1981.
4. *Kohonen T.* Self-Organizing Maps. 3d Extended Ed. Berlin; New York: Springer-Verlag, 2001.
5. *Fritzke B.* Unsupervised Ontogenic Networks. Handbook of Neural Computation / Eds.: Fiesler E., Beale R. Oxford Univ. Press, 1997.
6. *Skvortsov A. V.* Delaunay Triangulation and Its Applications. Tomsk: Tomsk State Univ. Publ., 2002.
7. *Cousty J. et al.* Watershed Cuts: Minimum Spanning Forests and the Drop of Water Principle // Pattern Analysis and Machine Intelligence, IEEE Trans. 2009. V. 31, No. 8. P. 1362; 1374.
8. *Bertrand J.* On the Dynamics // Image and Vision Computing. 2007. V. 25. P. 447–454.
9. *Mitsyn S. V., Ososkov G. A.* The Growing Neural Gas and Clustering of Large Amounts of Data // Optical Memory and Neural Networks. 2011. V. 20, Iss. 4. P. 260–270.

Received on May 30, 2014.