A Soft Clustering Algorithm Based on k-median

Ertuğrul Kartal Tabak
Computer Engineering Dept.
Bilkent University
Ankara, Turkey 06550
Email: tabak@cs.bilkent.edu.tr

Abstract

The k-median problem is one the powerful problems for clustering, despite its hard-to-resolve NP-hard property. In the k-median problem, we are given n nodes and some distance function, we are to select k of them to be cluster centers. The remaining nodes will then assigned to their nearest medians. The objective function is to minimize sum of distances. In fuzzy clustering, a node may be partially be included in more than two clusters. In some cases, it may be valuable to adopt existing k-median algorithms to fuzzy clustering problems.

In this paper, we define a *fuzzy* k-median problem for use in clustering problems. The problem fits most to the cases where some Bayesian model is expected. We show that the integer programming formulation of fuzzy k-median is simpler than original k-median problem. We also provide an approximation algorithm to the fuzzy k-median problem.

I. Introduction

Clustering is a widely used technique in unsupervised learning field of Machine Learning. Clustering is used especially for data analysis, data categorization, and fetching representatives for clusters of large data. Clusters may be thought as the process of producing unlabeled categorized data.

A. Terminology

Clustering can be defined as grouping similar objects together. The *similarity* can be defined in various ways. The most common definition of similarity is the distance function. The distance function may be the Euclidian distance or p-norm on metric space, or it can be the cosine-distance as in text categorization.

Metric System: A *metric* distance function is a function that obeys two rules: symmetry and triangle inequality. Some people add the reflexivity condition to the metric regulations, but generally, reflexivity condition can be relaxed by mapping data to some other metric system, and this condition

may not be considered as a must. p-norm is a metric function, but cosine-distance is not, since cosine-distance does not obey triangle inequality. On the other hand, cosine-distance obeys a relaxed version of triangle inequality, and cosine-distance is called a *nearly-metric* function.

Although, it is desirable to have a metric distance function; in some problems, the distance function does not satisfy the metric conditions. It may be the case that the distance function is not symmetric. It is possible to have a distance function that is not a real-valued function. Even it may be the case that it is not required the distance function to be defined over all the points.

Complexity: Generally clustering algorithms are not so simple, and most of them are NP-hard, i.e. they does not have polynomial time solutions unless P=NP. This phonemena created a research challenge on clustering algorithms. Some common algorithms of unsupervised learning and clustering are discussed in [4], [7] and [5]. In [6], Gonzales et al. proposed generic solutions to some domain of clustering problems.

Approximation. Clustering problems are generally NP-hard. Thus, it is a must to come up with approximations to the problem rather than the optimal solutions. If we have an α*-approximation* algorithm for a particular problem, that algorithm is said to guarantee that the cost of the result of the algorithm does not exceed α times the cost of the optimal solution of the problem.

Large DataSets: Clustering on large datasets has been given a special value of interest. Since most clustering problems are NP-hard, the input size becomes more important in the evaluation of an algorithm. Bradley et al. [1] discussed possible adoptations of clustering algorithms to large datasets. [9] presents a data clustering algorithm for large datasets, addressing I/O cost minimization. [8] proposed a sampling based $O(1)$-approximation algorithm with running time independent of data size.

Outliers: In case of there is few noise or very distant data in the training set, the algorithm may be required to discard a portion of data as noise. The data to be discarded are called *outliers*.

Soft Clustering: There are different notions of clustering, one of the interesting differenting notions is *soft* and *hard* clustering techniques. In *hard* clustering, each node should be assigned to exactly one median, whereas in *soft* clustering, it is more desirable to let a node to be assigned to several clusters partially. The soft clustering is also called *fuzzy* clustering, since the classification is not exact. Chiang et al. [3] discussed fuzzy classifiers and their relation with decision trees.

B. Some Clustering Algorithms and k-median

There are several clustering problems available. k-median is one of the most popular and powerful clustering problems. The definition of k-median is as follows: Given a data set N of nodes, a distance function $d : N^2 \to \mathcal{R}$, and an integer k; find a k element subset of N as *medians* such that sum of

distances from each node to its nearest median is minimal. The nodes that are closer to a median form a cluster. For any node, the node is said to be *assigned* to its nearest median.

k-median is a specialized version of *Uncapacitated Facility Location Problem*. In Uncapacitated Facility Location Problem, we also have a cost of declaring a node as a median (or *facility*). The optimization function is the cost function of k-median plus costs of opening facilities at the median nodes.

k-center problem is similar to the k-median problem. In the case of k-center, the objective function is to minimize the maximum of the distances of nodes to the assigned medians (or in that case *centers*) in the clusters.

k-means problem is another problem that is similar to the k-median problem. In the case of k-means, the objective function is the same as k-median but the nodes (or *points*) are not restricted to be elements of N.

C. Outline

In this paper, we will focus on the definition of fuzzy clustering for k-median algorithm. In Section II, we give the formal definition of k median. In Section III, we will define a fuzzy k-median algorithm. In Section IV we will present an approximation algorithm for fuzzy k-median algorithm. In the last Section we will summarize our work.

II. k-MEDIAN

As defined earlier, k-median is an optimization problem. Like many optimization problems, k-median can be expressed as an integer programming [2].

A. Terminology

Let N be the set of input data.

Let d_j be the demand of $j \in N$. This demand may be considered as the number of points at node j. In some problem definitions, d_j is not taken into account in the formulation, which can be formulated as letting $d_j = 1$ for all j.

Let c_{ij} be the cost of assigning $j \in N$ to the median $i \in N$.

Let x_{ij} be the final assignment of $j \in N$ to the median $i \in N$. It is a $0 - 1$ variable, and when $x_{ij} = 1$, it means that node j is assigned to i.

Let y_i be a 0-1 variable indicating whether node i is selected as a median.

B. Integer Programming Formulation

Using these definitions, the equation can be formulated as:

$$\text{minimize} \sum_{i,j \in N} d_j c_{ij} x_{ij} \tag{1}$$

subject to

$$\sum_{i \in N} x_{ij} = 1 \quad \forall j \in N \tag{2}$$

$$x_{ij} \leq y_i \quad \forall i, j \in N \tag{3}$$

$$\sum_{i \in N} y_i = k \tag{4}$$

$$x_{ij} \in \{0, 1\} \quad \forall i, j \in N \tag{5}$$

$$y_i \in \{0, 1\} \quad \forall i \in N \tag{6}$$

The constraints (2) ensure that each node is assigned to some other. The constraints (3) ensure that the assignment is made to a median. The constraints (4) ensure that there are k medians.

C. Related Work

Charikar et al. [2] presents an approximation algorithm based on this integer programming. They relax the x_{ij} and y_i to be in $[0, 1]$ and thus converting this integer programming problem to a linear programming problem. Then, based on the optimal solution of the LP, they round the solutions to satisfy the constraints (5) and (6). We will follow some similar way when we provide a solution to the fuzzy k-median problem. Their solution achieves a $6\frac{2}{3}$-approximation algorithm.

III. FUZZY k-MEDIAN

The fuzzy k-median problem is similar to the k-median problem, in the sense that there are k medians. But in that case, the nodes can be assigned to more than one medians. The assignment will be based on some probability as we have in Bayesian theory.

Let $P(m_i|j)$ be the probability of a node j to be in cluster m_i. Here, m_i is represented by its median, say i. As in Bayesian theory, we can write the following:

$$P(m_i|j) = \frac{P(j|m_i)P(m_i)}{P(j)} \tag{7}$$

Let $n = \sum_{j \in N} d_j$, the size of total demand. Now we can assume:

$$P(j) = \frac{d_j}{n} \tag{8}$$

For the sake of simplicity, let

$$P(m_i) = w_i \tag{9}$$

w_i corresponds to the weight of median i. Thus it is an output variable.

The following assumption will improve the validity of our model:

$$P(j|m_i) = f(c_{ij}) \quad \text{where } f \text{ is } o(1/c_{ij}) \tag{10}$$

Here (10) assumes that the probability function solely depands on the distance metric, and it decreases faster than c_{ij} increases. Note that, $P(j|m_i)$ is an input, neighter the function nor its values do not change by the output of the algorithm.

The objective function is minimize $\sum_{i,j \in N} P(m_i|j) d_j^2 c_{ij}$. As you might have noticed, we have replaced the x_{ij} part with the probabilistic behaviour.

We should also note that $P(m_i|j)$ is a probability function satisfying:

$$\sum_{i \in N} P(m_i|j) = 1 \quad \forall j \in N. \tag{11}$$

So, we will be ensured that total assignment of a node is 1.

Finally, define

$$q_{ij} = P(j|m_i) c_{ij} \tag{12}$$

$$q_i = \sum_{j \in N} q_{ij} d_j \tag{13}$$

Based on the assumptions, the minimization function becomes:

$$\sum_{i,j \in N} P(m_i|j) d_j^2 c_{ij} = \sum_{i,j \in N} \frac{P(j|m_i) P(m_i)}{P(j)} d_j^2 c_{ij} \tag{14}$$

$$= \sum_{i,j \in N} \frac{P(j|m_i) w_i}{\frac{d_j}{n}} d_j^2 c_{ij} \tag{15}$$

Since n does not depand on i or j, we can ignore it, and the objective function becomes:

$$\sum_{i,j \in N} \frac{P(j|m_i) w_i}{d_j} d_j^2 c_{ij} = \sum_{i,j \in N} P(j|m_i) w_i d_j c_{ij} \tag{16}$$

$$= \sum_{i,j \in N} w_i q_{ij} d_j \tag{17}$$

$$= \sum_{i \in N} w_i \sum_{j \in N} q_{ij} d_j \tag{18}$$

$$= \sum_{i \in N} w_i q_i \tag{19}$$

$$\tag{20}$$

Now we can formulate our problem:

$$\text{minimize} \sum_{i \in N} w_i q_i \tag{21}$$

subject to

$$\sum_{i \in N} y_i \quad = k \tag{22}$$

$$w_i \quad \leq y_i \quad \forall i \in N \tag{23}$$

$$y_i \quad \in \{0,1\} \quad \forall i \in N \tag{24}$$

$$w_i \quad \in [0,1] \quad \forall i \in N \tag{25}$$

This model is much simpler than the original integer programming model of k-median. In the original problem, there were $O(N^2)$ constraints, but in our model, this number of constraints have been dropped to $O(N)$. On the other hand, the output of a fuzzy k-median will also be an approximation of the original k-median problem.

IV. An Algorithm for Fuzzy k-median

Now, we will propose an approximation algorithm for fuzzy k-median.

As in [2], we will relax the integer programming constraints to linear programming constraints. this leads the following modification:

$$y_i \in \{0,1\} \forall i \in N \rightarrow y_i \in [0,k] \forall i \in N \tag{26}$$

Let C_j be the marginal cost of adding a new client to node j:

$$C_j = \sum_{i \in N} w_i q_{ij} \tag{27}$$

Note that optimal cost of linear programming $C_{LP} = \sum_{j \in N} d_j C_j$. The algorithm will solve this LP, and then round the solution to a feasible solution for the integer programming model of fuzzy k-median.

To round the fractional solution, the algorithm uses the one presented by Charikar et al. in [2].

Finally the computed w_i's are returned.

V. Conclusion

We have defined the fuzzy k-median problem for use in clustering problems. The problem fits most to the cases where some Bayesian model is expected. Although the fuzzy k-means problem is available for these kind of problems, there may be cases where the solution set should be a subset of the original. We have shown that the formulation of fuzzy k-median is simpler than original k-median problem. We have also provided an approximation algorithm to approximate fuzzy k-median problem.

Acknowledgment

The author would like to thank to his wife and child. They were in extreme understanding and help during preperation of this work.

References

[1] Paul S. Bradley, Usama M. Fayyad, and Cory Reina. Scaling clustering algorithms to large databases. In *Knowledge Discovery and Data Mining*, pages 9–15, 1998.

[2] M. Charikar, S. Guha, É. Tardos, and D. Shmoys. A constant-factor approximation algorithm for the k-median problem. *Journal of Computer and System Sciences*, 2002.

[3] I-Jen Chiang and Jane Yung jen Hsu. Integration of fuzzy classifiers with decision trees. In *Proc. Asian Fuzzy Syst. Symp.*, pages 65–78, Kenting, Taiwan, 1996.

[4] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification*. John Wiley and Sons, Inc., 2000.

[5] T. Feder and D. Greene. Optimal algorithms for approximate clustering. In *Proc. 20th ACM Symp. Theory of computing*, pages 434–444, 1988.

[6] T. Gonzales. Clustering to minimize the maximum inter-cluster distance. *Theoretical Computer Science*, 1985.

[7] J. A. Hartigan. *Clustering Algorithms*. John Wiley and Sons, Inc., 1975.

[8] Adam Meyerson, Liadan O'Callaghan, and Serge Plotkin. A k-median algorithm with running time independent of data size. *Machine Learning*, 2004.

[9] Tian Zhang, Raghu Ramakrishnan, and Miron Livny. BIRCH: an efficient data clustering method for very large databases. In *ACM SIGMOD International Conference on Management of Data*, pages 103–114, Montreal, Canada, June 1996.