Applied Mathematics and Computation 295 (2017) 1–15

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

A fuzzy SV-*k*-modes algorithm for clustering categorical data with set-valued attributes

Fuyuan Cao[a,*], Joshua Zhexue Huang[b], Jiye Liang[a]

[a] Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, School of Computer and Information Technology, Shanxi University, Taiyuan 030006, China
[b] College of Computer Sciences & Software Engineering, Shenzhen University, Shenzhen 518060, China

ARTICLE INFO

Keywords:
Categorical data
Set-valued attribute
Set-valued modes
Fuzzy *k*-modes
Fuzzy SV-*k*-modes

ABSTRACT

In this paper, we propose a fuzzy SV-*k*-modes algorithm that uses the fuzzy *k*-modes clustering process to cluster categorical data with set-valued attributes. In the proposed algorithm, we use Jaccard coefficient to measure the dissimilarity between two objects and represent the center of a cluster with set-valued modes. A heuristic update way of cluster prototype is developed for the fuzzy partition matrix. These extensions make the fuzzy SV-*k*-modes algorithm can cluster categorical data with single-valued and set-valued attributes together and the fuzzy *k*-modes algorithm is its special case. Experimental results on the synthetic data sets and the three real data sets from different applications have shown the efficiency and effectiveness of the fuzzy SV-*k*-modes algorithm.

1. Introduction

The *k*-means algorithm is one of the most popular and best-known algorithms for clustering numerical data [1,2]. However, a lot of data in real applications are described by categorical attributes. For example, gender, profession, title, and hobby of customers are usually defined as categorical attributes. Unlike numeric data, categorical values are discrete and unordered. The standard *k*-means clustering process cannot be directly applied to categorical data due to lacking of geometric properties. Huang [3] proposed a *k*-modes algorithm to cluster categorical data by modifying the standard *k*-means clustering process [4]. In the *k*-modes algorithm, Huang used the simple matching dissimilarity measure to compute the distance between two categorical objects and represented the center of a cluster with modes instead of means and gave a frequency-based method to update modes. In [5], Huang further presented a fuzzy *k*-modes algorithm that is the fuzzy version of the *k*-modes algorithm in the framework of the fuzzy *k*-means algorithm [6]. Because of their efficiency in clustering very large categorical data, the *k*-modes and fuzzy *k*-modes algorithms have been widely used in various applications [7–12].

For most of data mining algorithms, a table or matrix is usually used as an input. In this matrix, each row represents an object and each column is an attribute only having a value for each object [13]. However, in real applications, an object may take multiple values in some attributes. For example, many people have more than one hobby in questionnaire. Such a data representation is widespread in many domains, such as retails, insurances and telecommunications. A more general data representation is shown in Table 1.

* Corresponding author.
E-mail addresses: cfy@sxu.edu.cn (F. Cao), zx.huang@szu.edu.cn (J.Z. Huang), ljy@sxu.edu.cn (J. Liang).

http://dx.doi.org/10.1016/j.amc.2016.09.023
0096-3003/

Table 1
An example data set on questionnaire.

ID	Name	Sex	...	Title	Hobby
1	John	M		{CEO, Prof.}	{Sport, Music}
2	Tom	M		{CEO, Chair}	{Reading, Sport}
...	...	...	...	...	...
n	Katty	F		{Prof., Chair}	{Traveling, Music}

Without loss of generality, data in Table 1 can be formulated as follows. Suppose that $\mathbf{X} = \{X_1, X_2, \ldots, X_n\}$ is a set of n objects and each object is described by m attributes $\{A_1, A_2, \ldots, A_m\}$, where $X_i = (X_{i1}; X_{i2}; \ldots; X_{im})$ and $1 \le i \le n$. Let V^j be the domain values of the attribute A_j in $\mathbf{X}$ and V^{A_j} be the power set of V^j, if $X_{ij} \in V^{A_j}$, we call X_i as a set-valued object and A_j as a set-valued attribute.

To cluster $\mathbf{X}$, the most intuitive method is to convert $V^j (1 \le j \le m)$ into $|V^j|$ binary categorical attributes. The value 0 or 1 indicates the categorical value is absent or present [14]. Although transformation simplifies the representation of set-valued objects, this treatment unavoidably results in semantic information loss, especially in the understandability of clustering results. Moreover, as the number of categorical attributes increases, two set-valued objects are very likely to be similar even if the categorical values they contain are very different [15].

Different distance functions between two objects often result in different cluster structures in clustering algorithms. The attribute values of different set-valued objects usually overlap for a given attribute instead of equal or unequal. For example, the objects 1 and 2 in Table 1 have one overlapping value "CEO" for the attribute *Title*. It is only natural that the dissimilarity measure between two set-valued objects should be in the range of [0,1] instead of {0, 1} for a given attribute. Thus, inherent clusters probably overlap in a data set. The fuzzy *k*-modes algorithm has obtained better results in clustering data with overlapping clusters [9]. Moreover, the fuzzy partition matrix can provide more information to help users to determine the final clustering and to identify the boundary objects.

In this paper, we propose a fuzzy method to cluster objects with set-valued attributes. The main contributions of the paper are outlined as follows:

- We define the center of a cluster as set-valued-modes which is a set-valued object that minimizes the sum of the distance between each object in the cluster and the set-valued modes.
- We develop a way to obtain the fuzzy partition matrix and give a heuristic update way of cluster centers to minimize the objective function.
- We propose a fuzzy SV-*k*-modes algorithm which can partition data with single-valued and set-valued attributes together and the fuzzy *k*-modes algorithm is its special case.
- We analyze the influence of the fuzziness factor for the effectiveness of the fuzzy SV-*k*-modes algorithm.
- Experimental results on the synthetic and real data sets have shown the efficiency and effectiveness of the fuzzy SV-*k*-modes algorithm.

The rest of this paper is structured as follows. Section 2 reviews the hard and fuzzy *k*-modes algorithms. In Section 3, a fuzzy SV-*k*-modes algorithm is presented. In Section 4, we propose an algorithm to generate set-valued data and validate the scalability of the fuzzy SV-*k*-modes algorithm. In Section 5, we show experimental results on the three real data sets from different applications. We draw conclusions in Section 6.

2. The hard and fuzzy *k*-modes algorithms

In this section, we briefly review the *k*-modes [3] and fuzzy *k*-modes [5] algorithms, which have become a very popular technique in clustering categorical data. Both these two algorithms use the simple matching dissimilarity measure for categorical objects, modes instead of means for clusters. They use different methods to update modes in the clustering process for minimizing the objective function. In the *k*-modes algorithm, a mode is composed of the value that occurs most frequently in each attribute for a given cluster. In the fuzzy *k*-modes algorithm, each attribute value of a mode is given by the value that achieves the maximum of the summation of membership degrees in a given cluster. These modifications have removed the numeric-only limitation of the *k*-means and fuzzy *k*-means algorithms [16].

Let $X = \{x_1, x_2, \ldots, x_n\}$ be a set of n objects described by a set of m categorical attributes $\{A_1, A_2, \ldots, A_m\}$, where $x_i = (x_{i1}; x_{i2}; \ldots; x_{im})$ and $1 \le i \le n$. The simple matching dissimilarity measure between x_i and x_j is defined as

$$d(x_i, x_j) = \sum_{s=1}^{m} \delta(x_{is}, x_{js}), \tag{1}$$

where

$$\delta(x_{is}, x_{js}) = \begin{cases} 0, & if\ x_{is} = x_{js}. \\ 1, & otherwise. \end{cases} \tag{2}$$