Soft Subspace Clustering for High-Dimensional Data

Liping Jing
Hong Kong Baptist University, Hong Kong

Michael K. Ng
Hong Kong Baptist University, Hong Kong

Joshua Zhexue Huang
The University of Hong Kong, Hong Kong

INTRODUCTION

High dimensional data is a phenomenon in real-world data mining applications. Text data is a typical example. In text mining, a text document is viewed as a vector of terms whose dimension is equal to the total number of unique terms in a data set, which is usually in thousands. High dimensional data occurs in business as well. In retails, for example, to effectively manage supplier relationship, suppliers are often categorized according to their business behaviors (Zhang, Huang, Qian, Xu, & Jing, 2006). The supplier's behavior data is high dimensional, which contains thousands of attributes to describe the supplier's behaviors, including product items, ordered amounts, order frequencies, product quality and so forth. One more example is DNA microarray data.

Clustering high-dimensional data requires special treatment (Swanson, 1990; Jain, Murty, & Flynn, 1999; Cai, He, & Han, 2005; Kontaki, Papadopoulos & Manolopoulos., 2007), although various methods for clustering are available (Jain & Dubes, 1988). One type of clustering methods for high dimensional data is referred to as subspace clustering, aiming at finding clusters from subspaces instead of the entire data space. In a subspace clustering, each cluster is a set of objects identified by a subset of dimensions and different clusters are represented in different subsets of dimensions.

Soft subspace clustering considers that different dimensions make different contributions to the identification of objects in a cluster. It represents the importance of a dimension as a weight that can be treated as the degree of the dimension in contribution to the cluster. Soft subspace clustering can find the cluster memberships of objects and identify the subspace of each cluster in the same clustering process.

BACKGROUND

Finding clusters from subspaces of high dimensional data, subspace clustering pursues two tasks, identification of the subspaces where clusters can be found and discovery of the clusters from different subspaces, i.e., different subsets of dimensions. According to the ways with which the subsets of dimensions are identified, subspace clustering methods are divided into the following two categories. *Hard subspace clustering* determines the exact subsets of dimensions where clusters are discovered. Typical examples include PROCLUS, HARP and others. (Chakrabarti & Mehrotra, 2000; Yip, Cheung, & Ng, 2004 ; Parsons, Haque, & Liu, 2004). *Soft subspace clustering* considers that each dimension makes a different level of contribution to the discovery of clusters and the degree of contribution of a dimension to a cluster is represented as the weight of this dimension. The subsets of the dimensions with larger weights in a cluster form the subspace of the cluster. Typical examples include LAC, COSA, SCAD and others (Domeniconi, Papadopoulos, Gunopulos, & Ma, 2004; Frigui and Nasraoui, 2004; Friedman and Meulman, 2004; Chan, Ching, Ng, & Huang, 2004; Law, Figueiredo, & Jain, 2004).

The above subspace clustering methods have more or less three problems. Firstly, they are not scalable to large data (e.g., HARP, COSA). Large high dimensional data can not be well handled with them. Secondly, some use a projection method (e.g., PROCLUS), which makes the clustering results non-understandable.

Recovery of the original dimensions from the projected dimensions turns out to be difficult. Thirdly, some (e.g., SCAD, LAC) can not handle sparse data, which is a well-known phenomenon in real applications (Jing, Huang, & Ng, 2005).

MAIN FOCUS

This chapter is focused on a new soft subspace clustering method. This method determines the subspaces of clusters according to the contributions of the dimensions in discovering the corresponding clusters. The contribution of a dimension is measured by a weight that is assigned to the dimension in the clustering process. Every dimension contributes to the discovery of clusters, but the dimensions with larger weights identify the subspaces of the clusters (Jing, Ng, & Huang, 2007). The new soft subspace clustering algorithm is based on the *k*-means clustering process. Therefore, it can cluster large and high-dimensional sparse data.

The *k*-Means Algorithm

The *k*-means algorithm (MacQueen, 1967) is the mostly used clustering algorithm in data mining. Given a set of numeric objects X and an integer *k*, the *k*-means algorithm searches for a partition of X into *k* clusters that minimizes the sum of the within groups squared errors. This process is often formulated as the following minimization problem.

$$F(U,Z)=\sum_{l=1}^{k}\sum_{i=1}^{n}\sum_{j=1}^{m}u_{i,l}(x_{i,j}-z_{l,j})^2 \quad (1)$$

subject to

$$\sum_{l=1}^{k}u_{i,l}=1,\ 1\le i\le n,\ 1\le l\le k,\quad u_{i,l}\in\{0,1\}$$

where $U=[u_{i,l}]$ is an $n\times k$ partition matrix and $u_{i,l}$ = 1 indicates that the *i*th object is allocated to the *l*th cluster. $Z=[z_{l,j}]$ is a set of *k* vectors representing the centers of the *k* clusters.

Problem P can be solved by iteratively solving two sub minimization problems. One is to minimize $F(U,\hat{Z})$ with a given $\hat{Z}$ as constant by

$$u_{i,l}=1 \text{ , if } \sum_{i=1}^{m}(x_{i,j}-z_{l,j})^2\le\sum_{i=1}^{m}(x_{i,j}-z_{r,j})^2 \text{ for } 1\le r\le k, \quad (2)$$

$u_{i,l}$ = 0, otherwise.

The other is to minimize $F(\hat{U},Z)$ with a given partition matrix $\hat{U}$ by

$$z_{l,j}=\frac{\sum_{i=1}^{n}u_{i,l}x_{i,j}}{\sum_{i=1}^{n}u_{i,l}} \quad (3)$$

for $1\le l\le k$, and $1\le j\le m$.

The convergence of this k-means minimization process is proved in (Selim & Ismail, 1984).

One of the drawbacks of the *k*-means algorithm is that it treats all features equally in deciding the cluster memberships of objects. This is not desirable when dealing with high dimensional data with a large number of diverse features (dimensions). In such data, a cluster structure is often confined to a subset of features rather than the whole feature set. Inclusion of other features can only obscure discovery of the cluster structure in the clustering process. This drawback can be removed with a feature weighting technique that can identify the feature importance and help the *k*-means algorithm to find clusters in subset of features.

Feature Weights

Feature weighting for clustering is an important research topic in statistics and data mining (Modha & Spangler, 2003; Huang, Ng, Rong, & Li, 2005). The main purpose is to select important features in which a weight is assigned to a dimension for the entire data set. A weight can also be assigned to a feature in each cluster according to the feature importance in forming the corresponding cluster (Jing, Huang, & Ng, 2005; Jing, Ng, & Huang, 2007). In other words, a *k x m* weight matrix $V = [v_{l,j}]$ (*l* is the cluster index and *j* the feature index) is built in the clustering process. In this matrix, different clusters have different feature weight values. The weight value for a feature in a cluster is inversely proportional to the dispersion of the feature values from the center of the cluster. Therefore, the high weight value indicates a small dispersion of the feature values in the