Projected Clustering for Biological Data Analysis

Ping Deng
University of Illinois at Springfield, USA

Qingkai Ma
Utica College, USA

Weili Wu
The University of Texas at Dallas, USA

INTRODUCTION

Clustering can be considered as the most important unsupervised learning problem. It has been discussed thoroughly by both statistics and database communities due to its numerous applications in problems such as classification, machine learning, and data mining. A summary of clustering techniques can be found in (Berkhin, 2002).

Most known clustering algorithms such as DBSCAN (Easter, Kriegel, Sander, & Xu, 1996) and CURE (Guha, Rastogi, & Shim, 1998) cluster data points based on full dimensions. When the dimensional space grows higher, the above algorithms lose their efficiency and accuracy because of the so-called "curse of dimensionality". It is shown in (Beyer, Goldstein, Ramakrishnan, & Shaft, 1999) that computing the distance based on full dimensions is not meaningful in high dimensional space since the distance of a point to its nearest neighbor approaches the distance to its farthest neighbor as dimensionality increases. Actually, natural clusters might exist in subspaces. Data points in different clusters may be correlated with respect to different subsets of dimensions. In order to solve this problem, feature selection (Kohavi & Sommerfield, 1995) and dimension reduction (Raymer, Punch, Goodman, Kuhn, & Jain, 2000) have been proposed to find the closely correlated dimensions for all the data and the clusters in such dimensions. Although both methods reduce the dimensionality of the space before clustering, the case where clusters may exist in different subspaces of full dimensions is not handled well.

Projected clustering has been proposed recently to effectively deal with high dimensionalities. Finding clusters and their relevant dimensions are the objectives of projected clustering algorithms. Instead of projecting the entire dataset on the same subspace, projected clustering focuses on finding specific projection for each cluster such that the similarity is reserved as much as possible.

BACKGROUND

Projected clustering algorithms generally fall into two categories: density-based algorithms (Agrawal, Gehrke, Gunopulos, & Raghavan, 1998; Procopiuc, Jones, Agarwal, & Murali, 2002; Liu & Mamoulis, 2005; Ng, Fu, & Wong, 2005; Moise, Sander, & Ester, 2006) and distance-based algorithms (Aggarwal, Procopiuc, Wolf, Yu, & Park, 1999; Aggarwal & Yu, 2000; Deng, Wu, Huang, & Zhang, 2006; Yip, Cheung, & Ng, 2003; Tang, Xiong, Zhong, & Wu, 2007). Density-based algorithms define a cluster as a region that has a higher density of data points than its surrounding regions. Dense regions only in their corresponding subspaces need to be considered in terms of projected clustering. Distance-based algorithms define a cluster as a partition such that the distance between objects within the same cluster is minimized and the distance between objects from different clusters is maximized. A distance measure is defined between data points. Compared to density-based methods in which each data point is assigned to all clusters with a different probability, distance-based methods assign data to a cluster with probability 0 or 1. Three criteria (Yip, Cheung, & Ng, 2003) have been proposed to evaluate clusters: the number of data points in a cluster, the number of selected dimensions in a cluster, and the distance between points at selected dimensions.

PROCLUS (Aggarwal, Procopiuc, Wolf, Yu, & Park, 1999) is a typical distance-based projected clustering algorithm which returns a partition of the data points, together with sets of dimensions on which data points in each cluster are correlated by using Manhattan segmental distance. However, this algorithm loses its effectiveness when points in different dimensions have different variance. We propose our algorithm, IPROCLUS (Improved PROCLUS) based on the following enhancements. We propose the modified Manhattan segmental distance which is more accurate and meaningful in projected clustering in that the closeness of points in different dimensions not only depends on the distance between them, but also relates to the distributions of points along those dimensions. Since PROCLUS strongly depends on two user parameters, we propose a dimension tuning process to reduce the dependence on one of the user parameters. We also propose a simplified replacing logic compared to PROCLUS.

MAIN FOCUS

Our algorithm, IPROCLUS, which allows the selection of different subsets of dimensions for different clusters, is based on PROCLUS. Our algorithm takes the number of clusters k and the average number of dimensions l in a cluster as inputs. It has three phases: an initialization phase, an iterative phase, and a cluster refinement phase. The medoid for a cluster is the nearest data point to the center of the cluster. The detail of our algorithm can be found in (Deng, Wu, Huang, & Zhang, 2006).

Modified Manhattan Segmental Distance

Manhattan segmental distance is defined as $(\sum_{i\in D}|p_{1,i}-p_{2,i}|)/|D|$ in PROCLUS. In our algorithm, we propose the modified Manhattan segmental distance as the distance measure to improve accuracy. We find that the closeness of points in different dimensions not only depends on the distance between them, but also depends on the distributions of points along different dimensions. Therefore we define a normalization factor n_i for each dimension, which is the standard deviation of all points in a dataset along dimension i. The modified Manhattan segmental distance between x_1 and x_2 relative to dimension set D can be defined as: $(\sum_{i\in D}|p_{1,i}-p_{2,i}| / n_i)/|D|$.

Initialization Phase

In the initialization phase, all data points are first chosen by random to form a random data sample set S with size A×k, where A is a constant. Then S is chosen by a greedy algorithm to obtain an even smaller set of points M with size B×k, where B is a small constant. The greedy algorithm (Gonzalez, 1985) is based on avoiding choosing the medoids from the same cluster. Therefore, the set of points which are most far apart are chosen.

Iterative Phase

We begin by choosing a random set of k points from M. Then the bad medoids (Aggarwal, Procopiuc, Wolf, Yu, & Park, 1999) in the current best medoids set are iteratively replaced with random points from M until the current best medoids set does not change after a certain number of replacements have been tried.

In each iteration, we first find dimensions for each medoid in the set, and form the cluster corresponding to each medoid. Then the clustering is evaluated and the bad medoids in the current best medoids set are replaced if the new clustering is better.

In order to find dimensions, several terms need to be defined first. For each medoid m_i, δ_i is the minimum distance from any other medoids to m_i based on full dimensions. The locality L_i is the set of points within the distance of δ_i from m_i. $X_{i,j}$ is the average distance to m_i along dimension j, which is calculated by dividing the average distance from the points in L_i to m_i along dimension j by the normalization factor n_j. There are two constraints when associating dimensions to medoids. The total number of dimensions associated to medoids must be equal to $k\times l$. The number of dimensions associated with each medoid must be at least 2. For each medoid i, we compute the mean $Y_i = \left(\sum_{j=1}^{d} X_{i,j}\right)/d$, and the standard deviation $\sigma_i = \sqrt{\sum_j \left(X_{i,j} - Y_i\right)^2 / (d-1)}$ of the values $X_{i,j}$. Y_i represents the average modified Manhattan segmental distance of the points in L_i relative to the entire space. Thus $Z_{i,j} = \left(X_{i,j} - Y_i\right)/\sigma_i$ indicates how the average distance along dimension j associated with the medoid m_i is related to the average