Distance function selection in several clustering algorithms

LU Yu
College of Computer Science, Chongqing University, Chongqing 400030, P.R. China
Received 13 May 2003; revised 25 February 2004

Abstract: Most clustering algorithms need to describe the similarity of objects by a predefined distance function. Three distance functions which are widely used in two traditional clustering algorithms k-means and hierarchical clustering were investigated. Both theoretical analysis and detailed experimental results were given. It is shown that a distance function greatly affects clustering results and can be used to detect the outlier of a cluster by the comparison of such different results and give the shape information of clusters. In practice situation, it is suggested to use different distance function separately, compare the clustering results and pick out the “swing points”. And such points may leak out more information for data analysts.

Keywords: distance function; clustering algorithms; k-means; dendrogram; data mining

1. Introduction

Clustering can be loosely defined as the process of organizing objects into groups so that the members within the same group are similar to each other in some way and different from members of the other group. The clustering techniques have been studied extensively by the mid-1970’s. In recent years, as the increased ability to generate data and the great development of World Wide Web, the concept of “data mining” massive databases has led to a renewal of interest in clustering algorithms. There are three major techniques of clustering: partitioning-based clustering (k-means for example), hierarchy-based clustering and model-based clustering. Most of them use some kind of metric (or distance) for the measurement of similarity between objects. However, in most literature, how to choose an appropriate distance to describe the similarity between objects is not investigated thoroughly. In other words, the question whether different distances will affect clustering results or not should be answered. And if yes, how to choose a distance function when facing a practical problem?

In this paper, a particular discussion is conducted on the distance selection strategy over a few of most famous clustering algorithms. Detailed experimental results on both synthetic and real world data sets are also presented.

2. Background and related works

Clustering is an important research area in data mining. There are many different kinds of clustering techniques. Some of them [1-2] do not adopt distance functions. However, such methods suffer similar criticism for their poor efficiency or theoretical understanding [3]. In fact, distance based clustering algorithms are more mature both in theoretical research and practical implementation [4-7]. As some authors [8] stated, there are different distance function candidates that can be utilized, however, few described the difference among such distance functions.

Ergun Akleman proposed a novel idea to parameterize convex symmetric shapes as loci by distance functions[9]. Although the original application is shape modeling in CAD domain, it also provides a thread in the distance function selection problems and can be adopted in cluster shape detecting.

3. Distance function analysis

As mentioned above, the task of this work is to find out the role of distance function in several famous distance-based algorithms, or at least to get some rule in choosing such functions while solving the real world problems. Mainly k-means algorithm and hierarchical clustering algorithm are to be discussed. All such algorithms need a well-defined distance function to present the similarity among different points in the input sample data space.

Definition 1.1. Let X be a set, and \( \mathbb{R}^+ \) be the set of all non-negative real numbers and \( x, y \in X \), a function \( \rho : X \times X \rightarrow \mathbb{R}^+ \) is called distance function or metric, if it satisfies the following distance conditions:
1) \( \rho(x, y) \geq 0 \), \( \rho(x, y) = 0 \) if and only if \( x = y \);
2) \( \rho(x, y) = \rho(y, x) \);
3) \( \rho(x, y) \leq \rho(x, z) + \rho(z, y) \) for any \( z \in X \) (the triangle inequality).

There exist various distance functions such as
Minkowski distance, Hamming distance, Mahalanobis distance and Hausdorff-Besicovitch distance. This work deals with Minkowski distance that is particularly interesting and widely used, and is given by
\[
\rho_p(x,y) = \left[ \sum_{i=1}^{n} |x_i - y_i|^p \right]^{\frac{1}{p}}; \quad x, y \in \mathbb{R}^n; \quad n, p \in \mathbb{N}.
\]
(1)

The distance calculated by Eq.(1) with \( p = 1 \) is called Manhattan distance; that with \( p = 2 \) Euclidean distance; and with \( p = \infty \), Chebyshev distance. So, Eq.(1) is changed respectively for Manhattan distance into
\[
\rho_p(x,y) = \rho_1(x,y) = \sum_{i=1}^{n} |x_i - y_i|; \quad n, p \in \mathbb{N},
\]
(2)
for Euclidean distance,
\[
\rho_p(x,y) = \rho_2(x,y) = \left( \sum_{i=1}^{n} |x_i - y_i|^2 \right)^{\frac{1}{2}}; \quad n, p \in \mathbb{N},
\]
(3)
and for Chebyshev distance,
\[
\rho_p(x,y) = \rho_\infty(x,y) = \max_{i=1}^{n} |x_i - y_i|; \quad n \in \mathbb{N}.
\]
(4)

A basic operation in clustering algorithm is to compare the similarity between objects. In comparison the similarity between objects, the objects are often projected as points into an \( n \)-dimension space and the distances between them are calculated. Considering the process of assigning a point to different given cluster centers while clustering, it is assumed that such point is attracted by the given centers and will be obtained by the “nearest” one. So in a spatial area, the given centers’ ability to attract the other points can be greatly affected by the definition of distance.

For two given points \( A \) and \( B \), the plane is partitioned into two regions \( R_A \) and \( R_B \), such that all points in \( R_A \) are closer to \( A \) than to \( B \), and all points in \( R_B \) are closer to \( B \) than to \( A \). For convenience, those different borders are replaced with lines, approximately.

There exist only four partitions when the plane is partitioned by Chebyshev distance according to the positions of points \( A \) and \( B \), as can be seen in Fig.1. However, when the line connecting \( A \) with \( B \) is vertical or horizontal, there exist areas that cannot be partitioned, in other words, points belong to such area have the same distance to \( A \) and \( B \), which are called vague areas.

There also just exist four partitions of points \( A \) and \( B \) according to their positions when Manhattan distance is used to partition the plane, and vague areas appear when the angle between the line \( AB \) and one axis is \( 45^\circ \), as shown in Fig.2.

On contrast, when using Euclidean distance, the partition border is changed continuously with respect to the different positions of point \( A \) and \( B \). Thus there exist innumerable partitions. The border of two partitioned region is always orthogonal to the line connecting \( A \) with \( B \). And there never exist the vague areas.

Such conclusions can be generalized into three or even higher dimension metric space both by intuition (see Fig. 4) and mathematic proof.

![Fig.1. Partitioning the plane using Chebyshev distance](image1)

![Fig.2. Partitioning the plane using Manhattan distance](image2)

![Fig.3. Partitioning the plane using Euclidean distance](image3)

![Fig.4. Simple Locus with Minkowski distance in 3-dimensional space](image4)

4. Distance function selection strategy

4.1 k-means algorithm:

Input: the number of clusters \( k \) and \( m \) points in \( n \)-dimensional space.
Output: A set of $k$ clusters, satisfying some end criterion or error function.

Method:
1) Determine the $k$ initial cluster centers;
2) Repeat;
3) Assign each point to the cluster whose centre is nearest to it;
4) Update the cluster centre by calculating mean value of the points for each cluster;
5) End till no change.

4.2 Hierarchical algorithm

Input: $m$ vectors in $n$-dimensional space.

Output: a bin-tree named Dendrogram.

Method:
1) Initialize: Assign each vector to its own cluster;
2) Compute distances between all clusters;
3) Merge the two clusters that are closest to each other;
4) Return to step 2 until there is only one cluster left.

It is obvious that both algorithms are sensitive to the selection of distance function.

When implementing k-means, points are separated linearly by different distance function selection strategies. In practice, it is suggested to try both distance function to get more information about the distribution of the points, which can provide helpful information for outlier detection and the shape of clusters.

What is more, in implementing hierarchical clustering, it was found difficult to predict the results when choosing different distance function. There has been report on bad performance [10] of such clustering algorithm. Once engaged in large data sets, such cluster algorithm is not acceptable.

5. Experimental results

The experiments were performed by using data [2,10] obtained from a synthetic data generator and the real world data.

5.1 Synthetic data studies

A data generation program was designed to produce data sets with specific patterns using a combination of parameters, such as the data size, scope and shape. Several data sets were generated, which each consist of random distribution points in three elliptical areas. Fig.5 shows the different clustering results by k-means and hierarchical clustering algorithms.

5.2 Real world data studies

The famous Iris Data set was employed to study the distance function selection problem. The data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant. One class is linearly separable from the other 2; and the latter are not linearly separable from each other.

Fig.6 shows that k-means algorithm is obviously a linear clustering algorithm. It partitions the first cluster successfully, however, makes some mistakes in the latter
two that should not be linearly separable from each other. Hierarchical clustering algorithm can identify the first cluster too. The same results are obtained when the Euclidean and Chebykshev distances are selected by hierarchical algorithm, but quite different results appear when using Manhattan distance. It is hard to reason for such results. The suggestion is that in applying hierarchical clustering algorithm, Manhattan distance be not recommended for its unpredictable results. Anyhow, it is not recommended to use hierarchical clustering algorithm in practical problems for its poor performance.

6. Discussion and conclusions

As already proposed in many articles [3,10], there are many other clustering algorithms not mentioned in this paper. Further more, there are other distance functions not discussed in this paper too. Particularly, how to design a special distance function so as to control the clustering results is an interesting issue, and need be investigated more deeply. Such work is now underway and the results will soon be available.

The study presented on distance function selection with k-means and hierarchical clustering algorithms shows that the k-means algorithm is a kind of linear clustering algorithm that can identify linearly separated groups of objects. Based on the experimental results, a different distance function results in different clustering results. It provides a new way to detect the outlier of a cluster by the comparison of such different results. Further more, the different results provide some information about the shape of clusters. It also shows that the traditional hierarchical clustering algorithm is computer intensive, and the results are unpredictable and not easy to understand.

References