International Journal of Information Sciences and Application.
ISSN 0974-2255 Volume 5, Number 1 (2013), pp. 11-20

http://www.irphouse.com

An Unsupervised Technique for Statistical Data Analysis Using Data Mining

Dr. K. Meena[***], Dr. M. Manimekalai[**] and Mrs. S. Rethinavalli*

****Vice Chancellor, Bharathidasan Univer sity,*
Tiruchirappalli, Tamil Nadu, India.
***Director, Department of MCA, Shrimati Indira Gandhi College,*
Tiruchirappalli, Tamil Nadu, India.
** Assistant Professor, Department of MCA,Shrimati Indira Gandhi College,*
Tiruchirappalli, Tamil Nadu, India.

Abstract

Cluster analysis divides data into meaningful or useful groups (clusters). If meaningful clusters are the goal, then the resulting clusters should capture the "natural" structure of the data. For example, cluster analysis[1] can be used to group related documents for browsing, to find genes and proteins that have similar functionality, and to provide a grouping of spatial locations prone to earthquakes. However, in other cases, cluster analysis is only a useful starting point for other purposes, e.g., data compression or efficiently finding the nearest neighbors of points.

It is a main task of exploratory data mining, and a common technique for statistical data analysis used in many fields, including machine learning, pattern recognition, image analysis, information retrieval, and bioinformatics Cluster analysis methods are based on measuring similarity objects by computing distance between each pair. The scope of this paper is modest: to provide an introduction to cluster analysis in the field of data mining, where data mining has been defined. It is also found to be the discovery of useful, but non-obvious, information or patterns in large collections of data. Much of this paper is necessarily consumed with providing a general background for cluster analysis, but also a number of clustering techniques that have been recently developed specifically for data mining has also been discussed. This paper illustrates particular real-world applications.

Introduction

What is Cluster Analysis?

Cluster analysis groups objects (observations, events) based on the information found in the data describing the objects or their relationships. The goal is that the objects in a group will be similar (or related) to one other and different from (or unrelated to) the objects in other groups. The greater the similarity (or homogeneity) within a group, and the greater the difference between groups, the "better" or more distinct the clustering.

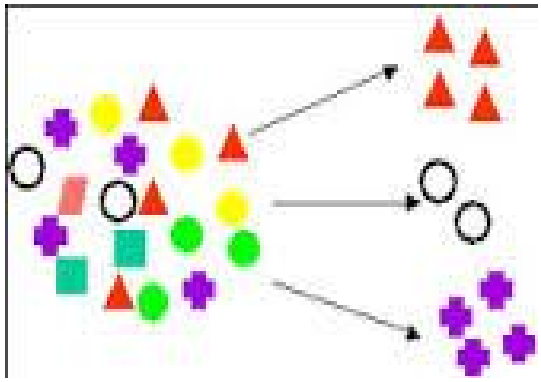


Figure 1.1

To better understand the difficulty of deciding what constitutes a cluster, consider Figures 1.1. Cluster analysis is a classification of objects from the data, where by classification we mean a labeling of objects with class (group) labels. As such, clustering does not use previously assigned class labels, except perhaps for verification of how well the clustering worked. Thus, cluster analysis is distinct from pattern recognition or the areas of statistics know as discriminate analysis and decision analysis, which seek to find rules for classifying objects given a set of pre-classified objects. While cluster analysis can be useful in the previously mentioned areas, either directly or as a preliminary means of finding classes, there is much more to these areas than cluster analysis. For example, the decision of what features to use when representing objects is a key activity of fields such as pattern recognition. Cluster Analysis typically takes the features as given and proceeds from there.

Distance Measures

The joining or tree clustering method [2] uses the dissimilarities or distances between objects when forming the clusters. These distances can be based on a single dimension or multiple dimensions. For example, if we were to cluster fast foods, we could take into account the number of calories they contain, their price, subjective ratings of taste, etc. The most straightforward way of computing distances between objects in a multi-dimensional space is to compute Euclidean distances. If we had a two- or three-dimensional space this measure is the actual geometric distance between

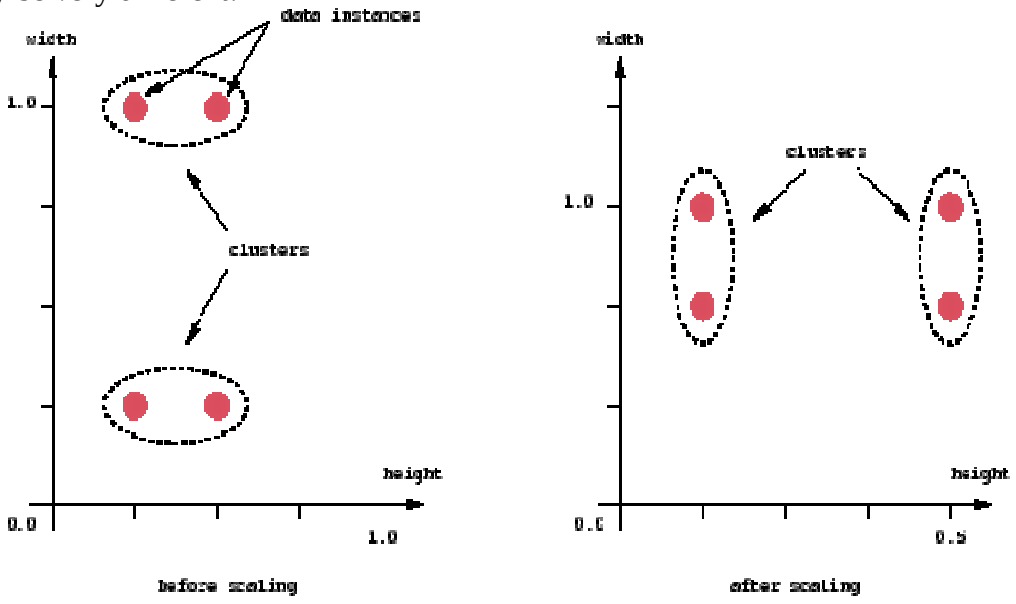
objects in the space (i.e., as if measured with a ruler). However, the joining algorithm does not "care" whether the distances that are "fed" to it are actual real distances, or some other derived measure of distance that is more meaningful to the researcher; and it is up to the researcher to select the right method for his/her specific application.

Euclidean Distance

This is probably the most commonly chosen type of distance. It simply is the geometric distance in the multidimensional space. It is computed as:

$$\text{Distance}(x,y) = \{ \Sigma_i (x_i - y_i)^2 \}^{½}$$

It is noted that Euclidean (and squared Euclidean) distances are usually computed from raw data, and not from standardized data. This method has certain advantages (e.g., the distance between any two objects is not affected by the addition of new objects to the analysis, which may be outliers). However, the distances can be greatly affected by differences in scale among the dimensions from which the distances are computed. For example, if one of the dimensions denotes a measured length in centimeters, and you then convert it to millimeters (by multiplying the values by 10), the resulting Euclidean or squared Euclidean distances (computed from multiple dimensions) can be greatly affected, and consequently, the results of cluster analyses may be very different.



Squared Euclidean distance

To square the standard Euclidean distance in order to place progressively greater weight on objects that are further apart, the distance is computed as (see also the note in the previous paragraph):

$$\text{Distance}(x,y) = \Sigma_i (x_i - y_i)^2$$

City-block (Manhattan) distance
This distance is simply the average difference across dimensions. In most cases, this distance measure yields results similar to the simple Euclidean distance. It is found that in this measure, the effect of single large differences (outliers) is dampened (since they are not squared). The city-block distance is computed as:

Distance (x,y)= $\Sigma_i |x_i - y_i|$

Chebychev distance
This distance measure may be appropriate in cases when one wants to define two objects as "different" if they are different on any one of the dimensions. The Chebychev distance is computed as:

Distance (x,y)= Maximum$|x_i - y_i|$

Power distance
To increase or decrease the progressive weight that is placed on dimensions on which the respective objects are very different, the power distance can be used which is computed as:

Distance(x,y)= (Σi |xi - yi|p)1/r

Where *r* and *p* are user-defined parameters. A few example calculations may demonstrate how this measure "behaves." Parameter *p* controls the progressive weight that is placed on differences on individual dimensions; parameter *r* controls the progressive weight that is placed on larger differences between objects. If *r* and *p* are equal to 2, then this distance is equal to the Euclidean distance.
Percent disagreement. This measure is particularly useful if the data for the dimensions included in the analysis are categorical in nature. This distance is computed as:

Distance(x,y)= (Number of $x_i \neq y_i$)/ i

Methods of Clustering

There are many clustering methods [3] available, and each of them may give a different grouping of a dataset. The choice of a particular method will depend on the type of output desired, The known performance of method with particular types of data, the hardware and software facilities available and the size of the dataset. In general , clustering methods may be divided into two categories based on the cluster structure which they produce. The non-hierarchical methods divide a dataset of N objects into M clusters, with or without overlap.

These methods are sometimes divided into *partitioning* methods, in which the classes are mutually exclusive, and the less common *clumping* method, in which overlap is allowed. Each object is a member of the cluster with which it is most similar, however the threshold of similarity has to be defined. The hierarchical methods produce a set of nested clusters in which each pair of objects or clusters is

progressively nested in a larger cluster until only one cluster remains. The hierarchical methods can be further divided into *agglomerative* or *divisive* methods.
In *agglomerative* methods, the hierarchy is build up in a series of N-1 agglomerations, or Fusion, of pairs of objects, beginning with the un-clustered dataset. The less common *divisive* methods begin with all objects in a single cluster and at each of N-1 steps divide some clusters into two smaller clusters, until each object resides in its own cluster.
Some of the important Data Clustering Methods are described below.

Partitioning Methods
The partitioning methods generally result in a set of M clusters, each object belonging to one cluster. Each cluster may be represented by a centroid or a cluster representative; this is some sort of summary description of all the objects contained in a cluster. The precise form of this description will depend on the type of the object which is being clustered. In case where real-valued data is available, the arithmetic mean of the attribute vectors for all objects within a cluster provides an appropriate representative; alternative types of centroid may be required in other cases, e.g., a cluster of documents can be represented by a list of those keywords that occur in some minimum number of documents within a cluster. If the number of the clusters is large, the centroids can be further clustered to produces hierarchy within a dataset.

Single Pass
A very simple partition method, the single pass method creates a partitioned dataset as follows:
1. Make the first object the centroid for the first cluster.
2. For the next object, calculate the similarity, S, with each existing cluster centroid, using some similarity coefficient.
3. If the highest calculated S is greater than some specified threshold value, add the object to the corresponding cluster and re determine the centroid; otherwise, use the object to initiate a new cluster. If any objects remain to be clustered, return to step 2.

As its name implies, this method requires only one pass through the dataset; the time requirements are typically of order *O(NlogN)* for order *O(logN)* clusters. This makes it a very efficient clustering method for a serial processor. A disadvantage is that the resulting clusters are not independent of the order in which the documents are processed, with the first clusters formed usually being larger than those created later in the clustering run

Hierarchical Agglomerative Methods
The hierarchical agglomerative clustering methods are most commonly used. The construction of an hierarchical agglomerative classification can be achieved by the following general algorithm.
1. Find the 2 closest objects and merge them into a cluster

2. Find and merge the next two closest points, where a point is either an individual object or a cluster of objects.
3. If more than one cluster remains , return to step 2

Individual methods are characterized by the definition used for identification of the closest pair of points, and by the means used to describe the new cluster when two clusters are merged.

There are some general approaches to implementation of this algorithm, these being stored matrix and stored data, are discussed below:

- In the second matrix approach , an N*N matrix containing all pairwise distance values is first created, and updated as new clusters are formed. This approach has at least an O(n*n) time requirement, rising to $O(n^3)$ if a simple serial scan of dissimilarity matrix is used to identify the points which need to be fused in each agglomeration, a serious limitation for large N.
- The stored data approach required the recalculation of pairwise dissimilarity values for each of the N-1 agglomerations, and the O (N) space requirement is therefore achieved at the expense of an $O(N^3)$ time requirement.

The Single Link Method (SLINK)

The single link method is probably the best known of the hierarchical methods and operates by joining, at each step, the two most similar objects, which are not yet in the same cluster. The name *single link* thus refers to the joining of pairs of clusters by the single shortest link between them.

The Complete Link Method (CLINK)

The complete link method is similar to the single link method except that it uses the least similar pair between two clusters to determine the inter-cluster similarity (so that every cluster member is more like the furthest member of its own cluster than the furthest item in any other cluster). This method is characterized by small, tightly bound clusters.

The Group Average Method

The group average method relies on the average value of the pair wise within a cluster, rather than the maximum or minimum similarity as with the single link or the complete link methods. Since all objects in a cluster contribute to the inter –cluster similarity, each object is , on average more like every other member of its own cluster then the objects in any other cluster.

Text Based Documents

In the text based documents, the clusters may be made by considering the similarity as some of the key words that are found for a minimum number of times in a document. Now when a query comes regarding a typical word then instead of checking the entire database, only that cluster is scanned which has that word in the list of its key words

and the result is given. The order of the documents received in the result is dependent on the number of times that key word appears in the document.

Data Mining Clustering

Techniques for Student data

K-means [4] is the simplest and most popular classical clustering method .The method is called K-means since each of the K clusters is represented by the mean of the objects called the centroid with in it. It is also called centroid method. The K –means method uses the Euclidean distance measure, which works well with compact clusters. If instead of Euclidean distance, the Manhattan distance is used the method is called the K-median method. In Data mining, *k*-medians clustering is a cluster analysis algorithm[5]. It is a variation of *k*-means clustering where instead of calculating the mean for each cluster to determine its centroid, one instead calculates the median. This has the effect of minimizing error over all clusters with respect to the 1-norm distance metric, as opposed to the square of the 2-norm distance metric (which *k*-means) does. consider the data about students in table 1..The only attributes are the age and the three marks.

Table.1: The seeds s1, s2, s3 are taken from the first three students from Table 1.

Student	**Age**	**Mark 1**	**Mark 2**	**Mark 3**
s1	18	73	75	57
s2	18	79	85	75
s3	23	70	70	52
s4	20	55	55	55
s5	22	85	86	87
s6	19	91	90	89
s7	20	70	65	60
s8	21	53	56	59
s9	19	82	82	60
s10	47	75	76	77

Table 2

Student	**Age**	**Mark 1**	**Mark 2**	**Mark 3**
s1	18	73	75	57
s2	18	79	85	75
s3	23	70	70	52

Computing the distance

Cluster analysis methods are based on measuring similarity between objects by computing the distance between each pair. Distance is a well understood concept that

has a number of simple properties.Distance is always positive,distance from point x to itself is always zero.Distance from point x to point y cannot be greater than the sum of the distance from x to some other point z and the distance from z to x.

Manhattan Distance

The distance is computed using the four attributes and using the sum of absolute difference using K-median method.

Let the distance between two points x and y (both vectors) be D(x,y)

$D(x,y)= \sum_i | x_i - y_i |$

Table 3: The distance values of all the objects is given in table 3.

c1	**18**	**73**	**75**	**57**	**Distance from clusters**			**Allocation to the nearest cluster**
c2	**18**	**79**	**85**	**75**	**From c1**	**From c2**	**From c3**	
c3	23	70	70	52				
s1	18	73	75	57	0	34	18	c1
s2	18	79	85	75	34	0	52	c2
s3	23	70	70	52	18	52	0	c3
s4	20	55	55	55	42	76	36	c3
s5	22	85	86	87	57	23	67	c2
s6	19	91	90	89	66	32	82	c2
s7	20	70	65	60	18	46	16	c3
s8	21	53	56	59	44	74	40	c3
s9	19	82	82	60	20	22	36	c1
s10	47	75	76	77	52	44	60	c2

The iteration leads to two students in the first cluster and four each in second cluster and third clusters.

Table 4: compares the clusters found in table 3. with the original seeds.

	Age	**Mark 1**	**Mark 2**	**Mark 3**
c1	18.5	77.5	78.5	58.5
c2	26.5	82.5	84.5	82.5
c3	21	61.5	61.5	56.5
s1	18	73	75	57
s2	18	79	85	75
s3	23	70	70	52

In the table 4 the mean marks for c3 are significantly lower than for c1 and c2.The new cluster means are used to recomputed the distance of each object to each of the means, again allocating each object to the nearest cluster.In the second iteration the number of students in cluster 1 is again 2 and the other two clusters still have 4

students each. The clusters have not changed at all. Therefore the method has converged rather quickly for this very simple dataset.

Table 5 present the average Euclidean distance of objects in each cluster to the cluster centroids. Therefore, the average distance within c1 of objects within it from its centroid is 5.9, while the average distance between objects in c2 and the centroid of c1 is 26.5.

Table 5.

Cluster	**c1**	**c2**	**c3**
c1	5.9	26.5	23.3
c2	29.5	14.3	42.6
c3	23.9	41	10.7

Distance between objects in c3 and the centroid of c1 is 23.3.These numbers show that the clustering method has done well in minimizing with-in cluster variance and maximizing between –cluster variance. These numbers do not show if there is another result that is better. These results will differ if last three seeds are taken. When we started with the last three objects as the seeds we obtained c1 with 4 students, with 5 students and c3 with only with 1 student. The one object in c3 was the starting seed for c3.Therefore one of the seeds is selected as an object that was an outlier and hence it appeared as a cluster of one object in the final result.

Conclusion

There are different clustering techniques in data mining that can be used for clustering student data. In this paper different distance measures and different types of clustering methods are mentioned. In future it is intended to improve the performance of these basic clustering of all attributes by converting it to similar scale to give more weight to some attributes that are relatively large in scale.

References

[1] Ritu Sharma(sachdeva), Afshar M Alam and Anita Rani. Article: K-Means Clustering in Spatial Data Mining using Weka Interface.IJCA Proceedings on International Conference on Advances in Communication and Computing Technologies.

[2] S. Sathya Bama, M.S.Irfan Ahmed and A.Saravanan. Article: Network Intrusion Detection using Clustering: A Data Mining Approach. International Journal of Computer Applications .

[3] Yujie Zheng+ Clustering Methods in Data Mining with its Applications in High Education. International Conference on Education Technology and Computer.

[4] RasimM. Alguliev, Ramiz M. Aliguliyev, and Saadat A. NazirovaClassification of Textual E-Mail Spam Using DataMining Techniques. Applied Computational Intelligence and Soft Computing.
[5] Bala Sundar V, T Devi and N Saravanan. Article: Development of a Data Clustering Algorithm for Predicting Heart. International Journal of Computer Applications.