CHAPTER 20

Cluster Analysis

20.Cluster Analysis

20.1 Introduction

Imagine that you are given a set of data objects for analysis where, unlike in classification, the class label of each object is not known. Clustering is the process of grouping the data into classes or clusters so that objects within a cluster have high similarity in comparison to one another, but are very dissimilar to objects in other clusters. Dissimilarities are assessed based on the attribute values describing the objects. Often, distance measures are used. Clustering has its roots in many areas, including data mining, statistics, biology, and machine learning.

Here you will learn the requirements of clustering methods for operating on large amounts of data. You will also study how to compute dissimilarities between objects represented by various attribute or variable types. You will study several clustering techniques, organized into the following categories: partitioning methods, hierarchical methods, density-based methods, grid-based methods, and model-based methods. Clustering can also be used for outlier detection.

20.2 What Is Cluster Analysis?

The process of grouping a set of physical or abstract objects into classes of similar objects is called clustering. A cluster is a collection of data objects that are similar to one another within the same cluster and are dissimilar to the objects in other clusters. A cluster of data objects can be treated collectively as one group in many applications. Cluster analysis is an important human activity. Early in childhood, one learns how to distinguish between cats and dogs, or between animals and plants, by continuously improving subconscious clustering schemes. Cluster analysis has been widely used in numerous applications, including pattern recognition, data analysis, image processing, and market research. By clustering, one can identify dense and sparse regions and, therefore, discover overall distribution patterns and interesting correlation among data attributes.

In business, clustering can help marketers discover distinct groups in their customer bases and characterize customer groups based on purchasing patterns. In biology, it can be used to derive plant and animal taxonomies, categorize genes with similar functionality, and gain insight into strictures inherent in populations. Clustering may also help in the identification of areas of similar land use in an

earth observation database, and in the identification of groups of automobile insurance policy holders with a high average claim cost, as well as the identification of groups of houses in a city according to house type, value, and geographical location. It can also be used to help classify documents on the web for information discovery. As a data' mining function, cluster analysis can be used as a stand-alone tool to gain insight I into the distribution of data, to observe the characteristics of each cluster, and to focus on a particular set of clusters for further analysis. Alternatively, it may serve as a preprocessing step for other algorithms, such as characterization and classification, which would then operate on the detected clusters. Data clustering is under vigorous development. Contributing areas of research include data mining, statics, machine learning, spatial database technology, biology, and marketing. Owing to the huge amounts of data collected in databases, cluster analysis has recently become a highly active topic in data mining research.

As a branch of statistics, cluster analysis has been studied extensively for many years, focusing mainly on distance- based cluster analysis. Cluster analysis tools based on k-means, k-medoids, and several other methods have also been built in to many statistical analysis software packages or systems, such as S-plus, SPSS, and SAS. In machine learning, clustering is an example of unsupervised learning. Unlike classification, clustering and unsupervised learning do not rely on predefine classes and class-labeled training examples. For this reason, clustering is a form of learning by observation, rather than learning by examples. In conceptual clustering, a group of objects forms a class only if it is describable by a concept. This differs from conventional clustering, which measures similarity based on geometric distance. Conceptual clustering consists of two components: (1) it discovers the appropriate classes, and (2) it forms descriptions for each class, as in classification. The guideline of striving for high intra class similarity and low inter class similarity still applies.

In data mining, efforts have focused on finding methods for efficient BE effective cluster analysis in large databases. Active themes of research focus the scalability of clustering methods, the effectiveness of methods for clustering complex shapes and types of data, high-dimensional clustering techniques and methods for clustering mixed numerical and categorical data in large databases.

20.3 Typical requirements

Clustering is a challenging field of research where its potential applications pose their own special requirements. The following are typical requirements clustering in data mining:

Scalability: many clustering algorithms work well on small data sets containing fewer than 200 data objects; however, a large database may contain millions of objects. Clustering on a sample of a given large data set may lead to biased results. Highly scalable clustering algorithms are needed.

Ability to deal with different types of attributes: many algorithms are designed to cluster interval-based (numerical) data. However, applications may require clustering other types of data, such as binary, categorical (nominal), and ordinal data, or mixtures of these data types.

Discovery of clusters with arbitrary shape: many clustering algorithms based on such distance measures tend to find spherical clusters with similar size and density. However, a cluster could be of any shape. It is important to develop algorithms that can detect clusters of arbitrary shape.

Minimal requirements for domain knowledge to determine knowledge to determine input parameters: many clustering algorithms require users to input certain parameters in cluster analysis (such as the number of desired clusters). The clustering results can be quite sensitive to input parameters. Parameters are often hard to determine, especially for data sets containing high dimensional objects. This not only burdens users,but also makes the quality of clustering difficult to control.

Ability to deal with noisy data: Most real-world databases contain outliers or missing,unknown,or erroneous data. Some clustering algorithms are sensitive to such data and may lead to clusters of poor quality.

Insensitive to the order of input records: Some clustering algorithms are sensitive to the order of input data; for example the same set of data,when presented with different orderings to such an algorithm,may generate dramatically different clusters.It is important to develop alorithms that are insensitive to the order of input.

High dimensionality: A database or a data warehouse can contain several dimensions or attributes.Many clustering algorithms are good at handling low-dimensional data,involving only two to three dimensions.Human eyes are good at judging the quality of clustering for up to three dimensions. It

is challenging to cluster data objects in high dimensional space, especially considering that such data can be very sparse and highly skewed.

Constraint-based clustering: Real-world applications may need to perform clustering under various kinds of consraints. Suppose that your job is to choose the locations for a given number of new automatic cash-dispensing machines(i.e.,ATMs) in a city.To decide upon this , you may cluster households while considering constraints such as the city's rivers and highway networks, and customer requirements per region. A challenging task is to find groups of data with good clustering behavior that satisfy specified consraints.

Interpretability and usability: Users expect clustering results to be interpretable, comprehensible,and sable. That is ,clustering may need to be tied up with specific semantic interpretations and applications. It is important to study how an application goal may influence the selection of clustering methods.

With these requirements in mind,our study of cluster analysis proceeds as follows.First,we study different types of data and how they can influence clustering methods.Second ,we present a general categorization of clustering methods.We then study each clustering method in detail, including partitioning methods, hierarchical methods,density-based methods,grid-based methods, and model-based methods. We also examine clustering in high-dimensional space and outlier analysis.

20.4 Types of Data in cluster Analysis

In this section,we study the types of data that often occur in cluster analysis and how to preprocess them for such an analysis.suppose that a data set to be clustered contains n objects, which may represent persons,houses ,documents, countries,and so on. Main memory-based clustering algorithms typecally operate on either of the following two data structures.

Data matrix(or object-by-variable structure): This represents n objects,such as persons,with p variables(also called measurements or attributes), such as age ,height,weight,gender,race,and so on.The structure is in the form of a relational table,or n-by-p matrix(nobjects x p variables).

$$\begin{bmatrix} x_{11} & \cdots & x_{1M} \\ x_{21} & \cdots & x_{2M} \\ \vdots & & \vdots \\ \vdots & & \vdots \\ x_{N1} & \cdots & x_{NM} \end{bmatrix}$$

Dissimilarity matrix(or object-by-object structure): This stores a collection of proximities that are available for all pairs of n objects. It is often represented by an n-by-n table:

$$\begin{bmatrix} 0 & & \\ d(2,1) & 0 & \\ \vdots & & \\ \vdots & & \end{bmatrix}$$

| d(n,1) d(n,2) ……0 |

-- ---

where d(i,j) is the measured difference or dissimilarity between objects i and j. In general , d(i,j) is a nonnegative number that is close to 0 when objects i and j are highly similar or "near" each other, and becomes larger the more they differ. Since d(i,j)===d(j,0),and d(i,i)=0.

The data matrix is often called a two-mode matrix, whereas the dissimilarity matrix is called a one-mode matrix, since the rows and columns of the former represent different entities,while those of the latter represent the same entity. Many clustering algorithms operate on a dissimilarity matrix.If the data are pre-sented in the form of a data matrix, it can first be transformed into a dissimilarity matrix before applying such clustering algorithms.

"How can -dissimilarity, d(i,j) be assessed?" you may wonder. In this section ,We discuss how object dissimilarity can be computed for objects described by interval-scaled variables;by nominal,ordinal,and ratio-scaled variables; or combinations of these variable types. The dissimilarity data can later be used to compute clusters of objects.

20.5 Interval-scaled Variables

This section discusses interval-scaled variables and their standardization.It then describes distance measures that are commonly used for computing the dissimilarity of objects described by such variables. These measures include the Euclidean, Manhattan, and Minkowski distances.

Interval-scaled variables are continuous measurements of a roughly linear scale Typical examples include weight and height, latitude and longitude coordinates (e.g..., when clustering houses), and

weather temperature.

The measurement unit used can affect the clustering analysis. For example, changing measurement unit used can affect the clustering analysis. For example, changing measurement units form meters to inches for height, or form kilograms to pounds for weight, may lead to a very different clustering structure.In general,expressing a variable in smaller units will lead to a larger range for that variable , and thus a larger effect on the resulting clustering structure. To help avoid dependence on the choice of measurement units, the data should be standardized. standardizing measurements attempts to give all variables an equal weight.This is particularly useful when given no prior knowledge of the data. However, in some applications, users may intentionally want to give more weight to a certain set of variables than to others.For example, when clustering basketball player candidates,we may prefer to give more weight to the variable height.

"How can the data for a variable be standardized?" To standardize measurements,one choice is to convert the original measurements to unitless variables. Given measurements f for a variable,this can be performed as follows.

1. calculate the mean absolute deviation ,sf.

$$Sf= (1\backslash n)(|x1f -mf|+|x2f-mf|+.....+|xnf-mf|)$$

where x1f,.....,xnf are measurements of f, and mf is the mean value of f, that is

$$mf=i=1/n(x1f+s2f+....+xnf)$$

2.calculate the standardized measurement,or z-score

$$z_{if} = x_{if} - m_f / s_f$$

The mean absolute deviation,sf is more robust to outliers than the standard deviation, of sf. When computing the mean absolute deviation, the deviation from the mean are not squared; hence, the effect of outliers i; somewhat reduced. There are more robust measures of despersion, such as the median absolute deviation.

However, the advantage of using the mean absolute deviation is that the z-scores of outliers do not become too small ; hence, the outliers remain detectable.

Standardization may or may not be useful in a particular application. Thus the choice of whether and how to perform standardization should be left to the user.

"OK," you now ask, "once I have standardized the data , how can I compute the dissimilarity between objects?" After standardization, or without standardization, or without standardization in certain applications, the dissimilarity (or similarity) between the objects described by interval-scaled variables is typically computed based on the distance between each pair of objects. The most popular distance measure is Euclidean distance, whih is defined as

$$D(I,j) = \sqrt{|X_{i1}-X_{j1}|^2+|X_{i2}-X_{j2}|^2+.....|X_{ip}-X_{ij}|^2}$$

Where $i = (X,_1 X,_2 .....X,p)$ and $j = (X_{j1}, X_{j2}.....X_{jp})$

Both the Euclidean distance and Manhattan distance satisfy the following

'mathematic requirements of a distance function';

1. d(i,j)>_0: Distance is a nonnegative number,

2. d(i,j)==0: The distance of an object to itself is 0.

3. d(i,j)=d(j,i): Distance is a symmetric function.

4. d(i,f)<d(i,h)+d(h,j): Going directly from object i to object j in space is no more than making a detour over any other object h(triangular in equalality).

Minkowski distance is a generqalization of oth Euclidean distance and Manhattan distance.It is defined as

$$d(i,j)=(|x_{1i}-x_{1j}|q+ |x_{i2}-x_{2j}|q+.....(|x_{ip}-xjp|q) 1/q$$

Where q is a positive integer.It represents the Manhattan distance when q=1,and Euclidean distance when q==2.

Weighting can also be applied to the Manhattan and Minkowski distances.

20.6 Binary Variables

This section describes how to compute the dissimilarity between objuces described by either symmeric or asymmetric binary variables.

A binary variable has only two states:0 or 1, where 0 means that the variable is absent,and 1 means that it is present. Given the variable smoker , describing a patient,for instance,1 indicates that the patient smokes, while 0 indicates that the patient does not . Treating binary variables as if they are interval-scaled can lead to misleading clustering results. Therefore, methods specific to binary data are necessary for computing dissimilariteis.

20.7 Nominal,Ordinal, and ratio-scaled Variables

This section discusses how to compute the dissimilarity between objects described by nominal, ordinal, and ratio- scaled variables.

Nominal Variables

A nominal variable is a generalization of the binary variable in that it can take on more than two states. For example , map color is a nominal variable that may have ,say,five states:red,yellow,green,pink,and blue.

Let the number of states of a nominal variable be M.The states can be denoted by letters, symbols,or a set of integers, such as 1,2,,..,M. Notice that such integers are used just for data handling and do not represent any specific ordering.

"How is dissimilarity computed between objects described by nominal variables?"

The dissimilarity computed between objects described by nominal variables?" The dissimilarity between two objects i and j can be computed using the simple matching approach:

$$d(i,j)=p-m/p$$

where m is the number of matches(i.e., the number of variables for which;'and j are in the same state), and p is the total number of variables, Weights can be assigned to increase the effect of m or to assign greater weight to the matches in variables having a large number of states.

Nominal variables can be encoded by asymmetric binary variables by creating a new binary variable for each of the M nominal states. For an object with a given state value, the binary variable representing that state is set to 1, while the remaining binary variables are set to 0. For example to encode the nominal variable map_color, a binary variable can be created for each of the five colors listed above. For an object having the color yellow, the yellow variable is set to 1, While the remaining four variables are set to 0.

20.8 Ordinal variables

A discrete ordinal variable resembles a nominal variable, except that the M states of the ordinal value are ordered in a meaningful sequence. Ordinal variables are very useful for registering subjective assessments of qualities that cannot be measured objectively . For example, professional ranks are often enumerated in a sequential order, such as assistant, associate, and full. A continuous ordinal variable looks like a set of continuous data of an unknown scale; that is , the relative ordering of the values is essential but their actual magnitude is not. For example the relative ranking in a particular sport(e.g., gold ,silver,bronze) is often more essential than the actual values of a particular measure. ordinal variables may also be obtained from the discretisation of interval-scaled quantities by splitting the value range into a finite number of classes.

The values of an ordinal variable can be mapped to ranks.For example,suppose that an ordinal variable has Mf states. These ordered states define the ranking 1,....,Mf

" How are ordinal variables handled?" The treatment of ordinal variables is , quite similar to that of interval-scaled variables when computing the dissimilarity between objects. Suppose that is a variable from a set of ordinal variables describing n objects. The dissimilarity computation with respect to involves teh following steps.

1. The value of f for the ith objet is xif, and has Mf ordered states, representing the ranking 1,....,Mf Replace each Xif , by its corresponding rank,

2. Since each ordinal variable can have a different number if states, it is often necessary to map the range of each variable onto[0.0,1.0] so that each variable has equal weight. This can be acheived by replacing the rand rid of the ith object in the fth variable by

zif =rif-1/mf-1

3.Dissimilarity can then be computed using any of the distance

20.9 Ratio-Scaled Variables

A ratio-scaled variable makes a positive measurement on a nonlinear scale,such as an exponential scale ,approximately following the formula

Ae pow Bt or Ae pow -Bt

where A and B are positive constraints. Typical examples include the growth of a bacteria population,or the decay of a radioactive element.

"How can I compute the dissimilarity between objects described by ratio-scaled variables?" There are three methods to handle ratio-scaled variables for computing the dissimilarity between objects.

- Treat ratio-scaled variables like interval-scaled variables.This ,however, is not usually a good choice since it is likely that the scale may be distorted.
- Apply logarithmic transformation to a ratio-scaled variable having values for object(by using the formulae -yif=log(xif). The yif values can be treated as interval-valued. Notice that for some ratio-scaled variables.log-log or other transformations may be applied.depending on the definition and application.
- treat xif as continuous ordinal data and treat their ranks as interval-valued.

The latter two methods are the most effective, although the choice of method used may be dependent on the given application.

20.10 Variables of mixed Types

We discussed how to compute the dissimilarity between objects described by variables of the same type, where these types may be either interval-scaled,symmetric binary , asymmetric binary,nominal,ordinal,or ratio-scaled.However,in many real databases, a mixture of variable types describes objects. In general ,a database can contain all of the sixvariable types listed above.

"so , how can we compute the dissimilarity between objects of mixed variable types?" One

approach is to group each kind of variable together ,performing a separate cluser analysis for each variable type.This is feasible if these analyses derive compatible results. However, in real applications,it is unlikely that a separate cluster

analysis per variable type will generate compatible results.

A more preferable approach is to process all variable types together,performing a single cluster analysis.One such technique combines the different variables into a single dissimilarity matrix.bringing all of the meaningful variables onto a common scale of the inteval [0,0,1,0].

20.11 Review questions

1. What Is Cluster Analysis?

2. Types of Data in cluster Analysis

3 Explain about Interval-scaled Variables

4 Discuss about Binary Variables

5 Discuss about Nominal,Ordinal, and ratio-scaled Variables

20.12 References.

[1]. Data Mining Techniques, Arun k pujari 1st Edition

[2] .Data warehousung,Data Mining and OLAP, Alex Berson ,smith.j. Stephen

[3].Data Mining Concepts and Techniques ,Jiawei Han and MichelineKamber

[4]Data Mining Introductory and Advanced topics, Margaret H Dunham PEA

[5] The Data Warehouse lifecycle toolkit , Ralph Kimball Wiley student Edition