Model for Clustering Objects under Specified Conditions

AMAURY A. CABALLERO, KANG K. YEN
Florida International University
JOSE L. ABREU
CJTech Corp.

Abstract. The problem of clustering objects under several conditions is frequently presented. The selection has been made in the past using statistical methods for discriminating certain parameters or creating queries from a database, which looks more practical. In general queries are created using the SQL method. The classical SQL methodology using crisp qualifiers causes difficulties in some decision making processes especially when it is mandatory to move to the definition of practical indicators or categories and to evaluate them according to certain practical assumptions. Recently fuzzy logic has been embedded in SQL to improve its performance, but the applications have been basically oriented to analysis of word similarity and indexing. The paper analyzes the statistical, SQL, and fuzzy SQL methods, presenting a practical application classifying several companies using the last one.

Key Words: Clustering, Data Bases, Classification, Modeling

1 Introduction

The objective of this paper is to analyze different methods for clustering objects, indicating their characteristics, and to show an easy-to-use methodology, which is based on fuzzy logic databases, which may improve the obtained results.

The problem of clustering objects under several conditions is frequently presented. The selection has been made using statistical methods for discriminating certain parameters or creating queries from a database, which looks more practical. In general queries are created using the SQL method. The classical SQL methodology using crisp qualifiers causes difficulties in some decision making processes especially when it is mandatory to move to the definition of practical indicators or categories and to evaluate them according to certain practical assumptions.

At the end, a case study is presented, related with the investment in companies that will represent a development for the investors and the society. The result can provide them with the information about the stage of the economic development among different companies. This classification can assist the investors make policies to boost the development of those companies that are lagging behind.

2 Methodologies

2.1 Statistical Methods

One of the exploratory methods in statistics is the factorial analysis. The principal components analysis (PCA) is one of the basic models of the factorial analysis when the objective is to predict the minimum number of necessary factors to justify the maximum portion of the variance represented in the original variables. By means of a mathematical procedure [1] a smaller

group of uncorrelated variables can be generated and called "principal components" that allows to identify a structure or the ownership from each individual to a specific group.

It is possible to complete the factorial analysis with a classification carried out on the total space or a sub-space defined by the first few significant factors. The classes consider the actual dimension of the cloud of points. The classification algorithms, particularly those of agglomeration, are locally robust since the low parts of the clusters (the nodes corresponding to the smallest distances) are independent of some isolated points [2]. This complementary character between the factorial analysis and the complete classification might complete the knowledge of data structure and allows a better interpretation of the data [3].

The inference and confirmatory statistics, however, allow us to validate the hypotheses formulated a priori (or after an exploratory phase) and to extrapolate the results to a wider population. This kind of statistics makes use of explanatory methods dedicated to explain and predict a variable starting from one or several explanatory variables, following the decision rules. Among these methods we find multiple and logistical regressions methods, discriminant analysis and analysis of the variance.

The main problem with all the statistical methods is the necessity of defining a priory some probability number, which makes these methods difficult to apply and obtain trustful results in many cases.

2.2 SQL Databases

For decades businesses and organizations have been busy accumulating vast stores of data. In the early 1970s, these businesses began to employ a new technology known as the relational database to produce useful information from the mountains of data they had accumulated. They began to organize their data into computerized warehouses known as databases. Several methods were devised to allow people to specify exactly what data they wished to retrieve from the database.

Although methods exist to facilitate the manipulation of data in a database, the most popular one is the declarative language known as Structured English Query Language (SQL). SQL can easily capture the mechanical intent of a query, but it lacks the ability to capture the semantics of a query. We can group and slice up collections of data in a variety of ways, but each division of the record collection proceeds along crispy lines [4]. For example, let's assume that the database "product" contains different entries with the product name and ID. A selection could be made through the following statement:

SELECT *PRODUCT ID*
FROM *PRODUCT*
WHERE *COST < 10*

In this case, the selected rows have a COST lower than 10. If the cost for some product is, for example, 10.1 units, this product will not appear in the query. This is an important limitation of SQL and this leads to one of the most significant barriers that people come up against when attempting to retrieve data from a database using SQL, which is the fact that they are forced to make arbitrary determinations about what does and does not fit the criteria they have in mind.

2.3. Fuzzy Databases

Since the introduction of the relational database model by Codd [5]many works [6] have proposed additional general database model, called fuzzy relational database models, to handle imprecise information. The literature reflects different ways of using fuzzy databases, but they are basically oriented to analysis of word similarity and indexing. Fuzzy search databases can be amassed that compile common misspellings (or variants) of specific words which can then be substituted during the cleansing process. This technique works better for applications that check one word at a time, like Microsoft® Word, which employs a similar technique for making spelling corrections on the fly. The Fuzzy Match advisor can be helpful in finding word variants or misspellings in a database-the result of typographical mistakes or errors during OCR (optical character recognition) processing. Also, if you have doubts about the spelling of a query word, you can use the Fuzzy Match Advisor to verify it.

In the present paper, the fuzzy model is used to obtain a query from the different tuples properties, as is normally done with SQL, but the properties (labels) have not crisp, but fuzzy boundaries.

An object in a database, represented by $(t_i, \mu_r(t_i))$ $(t, \mu_r(t))$, $(t_i, \mu_r(t_i))$ may or may not satisfy the select condition of a query [6]. Therefore, a tuple $(t, \mu_r(t))$ can belong to one of two components of a query: a satisfied part on an unsatisfied part. When $(t_i, \mu_r(t_i))$ satisfies the query, it belongs to the satisfied part; otherwise, it belongs to the unsatisfied part.

Let $(t, \mu_r(t)) = (t_i, \mu_r(t_i)) \vee \ldots \vee (t_k, \mu_r(t_k))$ Be a tuple in the extended fuzzy relation r, and let σ_φ be a query. Then

$$\text{Sat}(t) = \{(t_i, \mu_r(t_i)) / (\forall t_i)(t_i, \mu_r(t_i)) \in (t, \mu_r(t) \wedge (t_i, \mu_r(t_i)) \quad (1)$$

Satisfies the select condition of σ_φ.

Where Sat(t) represents the satisfied parts of $(t, \mu_r(t))$ for the query σ_φ.

The matching information refers to the matching degree provided by Sat(t) of tuple $(t, \mu_r(t))$. This matching degree can be expressed as the compatibility index (CI) defined by Cox [7].

Another possibility is to define the compatibility index using the average in place of the minimum value. The problem presented when using the degree of membership given by equation (1) is that if some of the degrees of membership $\mu_r(t_i)$ is under the α-cut threshold level, that tuple will not be included in the query results.

One thing that makes fuzzy systems useful is the ability to define "hedges," or descriptive modifiers, to represent fuzzy values. This keeps the operations of fuzzy logic closer to natural language and allows us to generate fuzzy statements through mathematical calculations.

Defining hedges and the operations that use them is a subjective process, and it can vary from project to project. But the system lets us use operators and produce compound results using the same formal methods as classic logic.

For example, let's change the statement "Company A is old" to "Company A is very old." Here we're using "very" as a hedge or descriptor, and this particular hedge is often defined as equivalent to the square of the base value. Therefore if OLD(Company A) = 0.90, then VERY OLD(Company A) = 0.81.

Other hedges include "more or less," "somewhat," "rather" and "sort of." All have subjective definitions but transform

membership/truth values in a systematic, reliable manner.

As stated by Buckes and Petry [8], "a tuple's membership value with respect to a query is a measure of the appropriateness of the tuple to the query. Therefore, query evaluation is the process to determine the truth value of a tuple to a query, given by the respective degrees of membership. When the matching degree of a tuple is less than the threshold value, that tuple is assumed to not satisfy the selected condition".

The method can be implemented through the following steps:

1) Define the utilized parameters (columns in the database).
2) Organize different regions for each parameter (Labels).
3) establish the range of variation of each parameter (universe of discourse) and the boundaries for each region (scope/domain)
4) Apply the selection algorithm (Crisp –Fuzzy).

Steps 1) through 3) may be used independently of the implemented selection algorithm. When crisp variables are used, SQL provides a straight forward solution, but with the previously mentioned restrictions.

The factor, which influences most on the performance in fuzzy logic application, is the definition of membership functions. The membership functions can be established as triangular functions having their maximum at the center of the membership function domain. Another issue to be considered is the α-cut threshold definition. Moving the α-cut threshold upwards, only highly compatible participants are selected [9]. Moving it down, a wider but less compatible set of rows is selected. In a first approximation the α-cut threshold can be selected as 0.1. The compatibility index obtained at the end of the process will represent by itself the more or less compatibility of the selected rows in the selection process.

2.3. Case Study: Analysis of Several Companies

The selected membership functions are triangular with 50% of overlapping. There are used only three membership functions for each parameter, equivalent to low, medium, and high. The twelve companies taken like examples are shown on Table 1.

A run for these companies, analyzing how much they fit the "Medium" condition, as per the software [10] is shown on Table 2 a). Note that the company "Industrial Material" is not selected. The reason is that its compatibility index is lower than 0.1. Any combination of conditions can be established. For example, if it is looking for a new company, with a high use of the machines, similar percent of male and female workers, with high qualifications, not taking into account the cost of the unitary product the result is shown on Table 2 b).

As can be seen, the selection indicates which companies are more or less compatible under the selected conditions. For example, under the first conditions the ropes factory was the less compatible (CI = 0.147), but it was the more compatible (CI = 0.633) for the second condition. In some cases, maybe desirable to use hedges to better represent the fuzzy values.

3 Conclusions

The paper analyzes several methods for clustering tuples, based on a group of previously defined indicators considered the priority and important for the evaluation.

The use of fuzzy logic databases is proved to result more adequate in many situations, because of its simplicity, accuracy and flexibility.

The extension of the model previously studied, provides a refinement that allows the user of the basic model to better reflect their concerns practical applications. The power of the fuzzy logic model is that it uses imprecise terms to arrive at 'crisp' values. Modifying these 'crisp' values by establishing weights, reflecting the importance of various attributes, is a logical next step.

The compatibility index (*CI*) gives a good measure in so far as a solution coincides with the previously imposed conditions. If the compatibility index is too low for any of the possible selections, it is necessary to revise the created fuzzy logic model: number and domain of the membership functions for each attribute. The results show that the fuzzy logic selection indicates which companies are more or less compatible with the selected conditions. If other factors are taken into consideration, the results will be different, depending on the input assumptions that can be changed, depending on the real situation under consideration. With this method is possible not only to have included in the query entries that using the classical SQL may not be included, but also organize them according to their more or less compatibility with the established conditions.

References

[1] Anderson, T.W., *An Introduction to Multivariate Statistical Analysis*, John Wiley, New York, 2nd edition,1984

[2] Sokal, R.R. & Sneath, P.H.A., *Principles of Numerical Taxonomy*, Freeman and Co., San Francisco, 1963

[3] Lebart, L., Morineau, A., & Poiron, M., *Statistique exploratoire multidimensinnelle,* Dunod, Paris, 1995

[4] Caballero, A.*, Construction Project Management Using Fuzzy Logic*, Journal of Information. Vol 6, No. 4, October 2003. Japan. pp 463 – 474.

[5] Codd, E. F., *Extending the Database Relational Model to Capture more Meaning.* ACM Transactions Database Systems. Vol.4, 1979, pp 397-434.

[6]. Wang, Ming-Hua et al., *A study of disjunctive Information in Fuzzy Relational Databases.* Journal of Information Sciences and Engineering, Vol. 22, 2006, pp 199-213.

[7] Cox, E.D*., Fuzzy Logic for Business and Industry*, Rockland: Charles River Media, 1995

[8]. Buckles, B. P., F. E. Petry, *A Fuzzy Representation of Data for Relational Databases.* Fuzzy Sets and Systems, Vol. 7, No. 3 1982, pp.213-226.

[9] Caballero, A.A., & Dye, J.M., *Comparison of Construction Firms Based on Fuzzy Set*s, J. of Construction Education, Vol.III, No.3, 1999, pp.305-312

[10] *Fuzzy Query 1.0*. (1998), Sonalysts, Inc., 215 Parkway North, Waterford, CT. 06385

Table 1. Companies Taken in the Study

Product	Starting Date	Cost $	% of use of Machines	% Women	Workers Qualification
Military Shoes	1945	160	50	10	6
Shirts	1992	120	75	90	8
Underwear	2001	30	90	80	10
Industrial Material	1922	780	100	1	10
Men's Shirts	1996	65	70	80	9
Shields	1998	35	80	50	8
Sport Forms	1989	200	70	85	10
Ropes	1998	55	100	80	10
Pants	1994	150	85	90	8
Industrial Forms	1971	80	80	85	7
Corsetiere	1965	60	70	62	10
Work Forms	1935	150	100	90	7

Table 2. a) Obtained Query for all the Parameters taken as "Medium"
b) Obtained Query for "High" percent of use of machinery; "Medium" percent of women; a "High" workers qualifications; and "New" company.

Company	Compatibility Index (CI)
Shields	0.61
Industrial Forms	0.60
Pants	0.49
Shirts	0.48
Corsetiere	0.44
Work Forms	0.41
Military Shoes	0.38
Sport Forms	0.36
Men's Shirts	0.34
Underwear	0.21
Ropes	0.14

a)

Company	Compatibility Index (CI)
Ropes	0.63
Underwear	0.52
Shields	0.47
Work Forms	0.41
Military Shoes	0.38
Industrial Material	0.35
Men's Shirts	0.34
Pants	0.31
Sport Forms	0.31
Corsetiere	0.3
Shirts	0.24
Industrial Forms	0.13

b)