HIDING FUZZY ASSOCIATION RULES IN QUANTITATIVE DATA

BY

K. SATHIYA PRIYA
0720108016

OF

KUMARAGURU COLLEGE OF TECHNOLOGY
COIMBATORE - 641006

A PROJECT REPORT

Submitted to the

FACULTY OF INFORMATION AND COMMUNICATION ENGINEERING

In partial fulfillment of the requirements
For the award of the degree

OF

MASTER OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

MAY 2009

BONAFIDE CERTIFICATE

Certified that this project report entitled "**HIDING FUZZY ASSOCIATION RULES IN QUANTITATIVE DATA**" is the bonafide work of **Ms. K. SATHIYAPRIYA**, who carried out the research under my supervision. Certified further, that to best of my knowledge the work reported herein is not from any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

Signature of the Guide
Mrs. N. Rajathi, M.E.,
Assistant Professor,
Department of Information
Technology.

Head of the Department
Dr.S.Thangasamy, Ph.D.,
Dean and Professor,
Department of computer
Science and Engineering

The candidate with **University Register No. 0720108016** was examined by us in the project viva-voce examination held on 5 · 5 · 09

5/5/09
INTERNAL EXAMINER

5/5/09
EXTERNAL EXAMINER

SNS COLLEGE OF TECHNOLOGY

Approved by AICTE and Affiliated to Anna University

(An ISO 9001:2000 Certified Institution)

Department of Information Technology

&

Society for Information Sciences and Computing Technology

Third National Conference on

NETWORKS IN [illegible] ENCE AND COMPUTING SYSTEMS

This is to certify that [illegible] Prof./M[illegible] [illegible]ya Priya M.E. CSE........

[illegible] has presented a technical paper

titled [illegible] tive Data Using Fuzzy Set Theory..........

in the [illegible] held on 16th February 2009.

Convenor

Prof.L.M.Nithya

Principal

Dr.V.P.Arunachalam

Director cum Secretary

Dr.S.N.Subbramanian

ABSTRACT

The recent advance of data mining technology to analyze vast amount of data has played an important role in marketing business. Despite its benefits in such areas, data mining also opens new threats to privacy and information security if not used properly. The main problem is that from non-sensitive data, one is able to infer sensitive information, including personal information, fact or even patterns that are not supposed to be disclosed.

Personal privacy problems are considered more seriously. Many techniques have been recently developed to prevent mining sensitive rules. These techniques are classified into two major categories. They are input privacy and output privacy. In input privacy, the data is manipulated, and the mining result is not affected or minimally affected. In output privacy, specific rules that should be hidden are given in advance and the data is altered, so that only non sensitive rules can be mined. This change makes the mining result preserve certain privacy. According to this constraint, many data altering techniques for hiding association, classification and clustering rules have been proposed. However, almost all of them have been done on binary items. But, in real world, the data mostly consist of quantitative values.

In this Project work, a new method that combines the quantitative association rule mining and fuzzy logic is used to hide sensitive association rules from quantitative data. For this purpose, support value of Left Hand Side (LHS) of the rule to be hidden is increased. The system is implemented to secure the breast cancer data set for patients. Experiments demonstrate that the results of the proposed algorithm will be consistent, will not miss any rule at the interval border.

ஆய்வுச்சுருக்கம்

தகவல் தேடல் தொழில்நுட்பத்தின் முன்னேற்றமானது இன்றைய வர்த்தகத்தில் முக்கிய பங்கு வகிக்கின்றது.

அவ்வாறாயினும், தகவல் தேடல் தொழில்நுட்பத்தினால் தகவல் பாதுகாப்பு தன்மையானது குறைகிறது.முக்கியத்துவம் அல்லாத தகவிலினை தேடும் பொழுது முக்கியத்துவம் வாய்ந்த தகவல்கள் வெளியாகி விடுகின்றது.

இத்தகைய தேடல்கள் தனித்துவ தகவல்களை தீண்டாமலிருக்க வரையறுக்கப்பட வேண்டும். இத்தகைய தொழில்நுட்பங்கள் இரு வகையாக பிரிக்கப்படுகின்றன. முதலாவதாக தகவல் தேடும்பொழுதே ஆராய்தல், அடுத்தது விதிகள் அமைத்து அதன் பலனாக முக்கியத்துவம் வாய்ந்த தகவல்களை தேடாமல் இருப்பது. இவ்வாறு விதிகளை மாற்றி அமைப்பது தகவல்களுக்கு ஓரளவு பாதுகாப்பினை அளிக்கின்றது. இத்தகைய கட்டுபாடுகள் தகவல் பிரிப்பு, தகவல் ஒருமைப்பாடு, தகவல் மறைப்பு போன்றவகைகளுக்கான விதிகளை உருவாக்குகின்றன.

இந்த ஆய்வு தொகுப்பில், திடகாத்திரமான தேடல் மற்றும் இருவெண்நிலை முறை மூலம், முக்கியத்துவம் வாய்ந்த தகவல்களை மறைக்கும் முறை கையாளப்பட்டுள்ளது. இதன் காரணமாக இடவிதியின் தாங்கு நிலை மதிப்பினை அதிகரித்து மறைக்கும் முறை பயன்படுத்தப்பட்டுள்ளது. மார்பு புற்று நோய் நோயாளிகளிடம் இருந்து கிடைத்த தகவல்களில் இத்திட்டத்தை பயன்படுத்தி சோதனை செய்ததில் வரையறுத்த எல்லையில் முரனில்லா விதியாக அமைவதாக உள்ளது.

ACKNOWLEDGEMENT

I express my profound gratitude to our Chairman **Padmabhusan Arutselvar Dr. N. Mahalingam B.Sc, F.I.E** for giving this opportunity to pursue this course.

I thank, **Dr. Joseph V.Thanikal, Ph.D.,** Principal, and **Prof. R. Annamalai,** Vice Principal, Kumaraguru College of Technology, Coimbatore, for being a constant source of inspiration and providing me with the necessary facilities to work on this project.

I would like to express a special acknowledgement and my honest thanks to **Dr. S. Thangasamy, Ph.D.,** Professor and Dean of Department of Computer Science and Engineering, for his support and encouragement throughout the project.

I convey my special thanks to **Ms.V.Vanitha, M.E.,** Assistant Professor and Project Coordinator, Department of Computer Science and Engineering for her valuable suggestions and guidance.

I express my deep sense of gratitude and gratefulness to my Project Guide, **Ms. N. Rajathi, M.E.,** Assistant professor, Department of Information Technology, for her kind support, supervision, tremendous patience, active involvement and guidance.

I would like to convey my honest thanks to all **members of staff** of the Department for their unlimited enthusiasm, friendship and experience from which I have greatly benefited.

I express my profound gratitude to my **parents, husband, kid and friends** for their moral support.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

ABBREVIATIONS	EXPANSION
KDD	Knowledge Discovery in DataBases
RHS	Right Hand Side
LHS	Left Hand Side
EDA	Exploratory Data Analysis
Lk	Set of frequent k-item sets(k-itemsets with minimum support)
Ck	Set of candidate k-itemsets (potentially frequent itemsets)
UkLk	Set of generated itemsets
ANFIS	Adaptive Neuro-Fuzzy Inference Systems
ISL	Increase the Support of Left handside
FTDA	Fuzzy Transaction Data Mining Algorithm
UCI	University of California, Irvine

LIST OF FIGURES

LIST OF TABLES

1. PROBLEM DEFINITION

Privacy preserving in data mining means hiding output knowledge of data mining by using several methods when this output data is valuable and private.

Most of the methods that prevent mining sensitive association rules were done on binary values. However, in real world, the data mostly consist of quantitative values. In quantitative association rule mining, the range of quantitative value of an attribute is divided into intervals. so it is possible to miss rules at the interval borders. The fuzzy set theory introduced by Zadeh provides excellent means to model the "fuzzy" boundaries by introducing gradual membership.

In this work, a method for preventing extraction of critical association rules from quantitative data is proposed, by combining Increase the Support of Left hand side (ISL) method that prevents extraction of critical association rules and fuzzy set theory.

2. INTRODUCTION

2.1 Introduction to Data Mining [13, 19]:

Data mining (sometimes called data or knowledge discovery) is the process of analyzing data from different perspectives and summarizing it into useful information – information that can be used to increase revenue, cut costs, or both. Data mining software is one of a number of analytical tools for analyzing data. It allows users to analyze data from many different dimensions or angles, categorize it, and summarize the relationships identified. Technically, data mining is the process of finding correlations or patterns among dozens of fields in large relational databases.

2.1.1 Data, Information, and Knowledge:

Data:

Data are any facts, numbers, or text that can be processed by a computer. Today, organizations are accumulating vast and growing amounts of data in different formats and different databases. This includes:

- Operational or transactional data such as, sales, cost, inventory, payroll, and accounting
- Non-operational data, such as industry sales, forecast data, and macro economic data
- Meta data – data about the data itself, such as logical database design or data dictionary definitions

Information:

The patterns, associations, or relationships among all this data can provide information. For example, analysis of retail point of sale transaction data can yield information on which products are selling and when.

Knowledge:

Information can be converted into knowledge about historical patterns and future trends. For example, summary information on retail supermarket sales can be analyzed in light of promotional efforts to provide knowledge of consumer buying behavior. Thus, a manufacturer or retailer could determine which items are most susceptible to promotional efforts.

2.1.2 Data Warehouses:

Dramatic advances in data capture, processing power, data transmission, and storage capabilities are enabling organizations to integrate their various databases into data warehouses. Data warehousing is defined as a process of centralized data management and retrieval. Data warehousing represents an ideal vision of maintaining a central repository of all organizational data. Centralization of data is needed to maximize user access and analysis. Dramatic technological advances are making this vision a reality for many companies. And, equally dramatic advances in data analysis software are allowing users to access this data freely. The data analysis software is what supports data mining.

2.1.3 What can data mining do?

Data mining is primarily used today by companies with a strong consumer focus – retail, financial, communication, and marketing organizations. It enables these companies to determine relationships among "internal" factors such as price, product positioning, or staff skills, and "external" factors such as economic indicators, competition, and customer demographics. And, it enables them to determine the impact on sales, customer satisfaction, and corporate profits. Finally, it enables them to "drill down" into summary information to view detail transactional data.

With data mining, a retailer could use point-of-sale records of customer purchases to send targeted promotions based on an individual's purchase history. By mining demographic data from comment or warranty cards, the retailer could develop products and promotions to appeal to specific customer segments.

For example, Blockbuster Entertainment mines its video rental history database to recommend rentals to individual customers. American Express can suggest products to its cardholders based on analysis of their monthly expenditures.

2.1.4 How does data mining work?

Data mining software analyzes relationships and patterns in stored transaction data based on open-ended user queries.

Data mining consists of five major elements:

- Extract, transform, and load transaction data onto the data warehouse system.
- Store and manage the data in a multidimensional database system.
- Provide data access to business analysts and information technology professionals.
- Analyze the data by application software.
- Present the data in a useful format, such as a graph or table

2.2 Data Mining Techniques [14] :

The ultimate goal of data mining is prediction. Predictive data mining is the most common type of data mining and one that has the most direct business applications. The process of data mining consists of three stages: (1) Initial exploration (2) Model building with validation/verification (3) Deployment

2.2.1 Exploration:

This stage usually starts with data preparation which may involve cleaning data, data transformations, selecting subsets of records and – in case of data sets with large numbers of variables ("fields") – performing some preliminary feature selection operations to bring the number of variables to a manageable range . Then, depending on the nature of the analytic problem, this first stage of the process of data mining may involve anywhere between a simple choice of straightforward predictors for a regression model, to elaborate exploratory analyses using a wide variety of graphical and statistical methods, in order to identify the most relevant variables and determine the complexity and/or the general nature of models that can be taken into account in the next stage.

2.2.2 Model Building and Validation:

This stage involves considering various models and choosing the best one based on their predictive performance. There are a variety of techniques developed to achieve that goal – many of which are based on so-called "competitive evaluation of models". This means that applying different models to the same data set and then comparing their performance to choose the best. These techniques – which are often considered the core of predictive data mining – include: Bagging (voting, averaging), Boosting (to generate multiple models or classifiers (for prediction or classification), and to derive weights to combine the predictions from those models into a single prediction or predicted classification), Stacking and Meta-Learning (to combine the predictions from multiple models. It is particularly useful when the types of models included in the project are very different. In this context, this procedure is also referred to as Stacking (Stacked Generalization)).

2.2.3 Deployment:

The final stage involves using the model selected as best in the previous stage and applying it to new data in order to generate predictions or estimates of the expected outcome.

The concept of Data Mining is becoming increasingly popular as a business information management tool where it is expected to reveal knowledge structures that can guide decisions in conditions of limited certainty. Data Mining is still based on the conceptual principles of statistics including the traditional Exploratory Data Analysis (EDA) and modeling and it shares with them both some components of its general approaches and specific techniques.

However, an important general difference in the focus and purpose between Data Mining and the traditional Exploratory Data Analysis (EDA) is that Data Mining is more oriented towards applications than the basic nature of the underlying phenomena. In other words, Data Mining is relatively less concerned with identifying the specific relations between the involved variables. For example, uncovering the nature of the underlying functions or the specific types of interactive, multivariate dependencies between variables are not the main goal of Data Mining. Instead, the focus is on producing a solution that can generate useful predictions. Therefore, Data Mining accepts among others, a "black box" approach to data exploration or knowledge discovery and uses not only the traditional Exploratory Data Analysis (EDA) techniques, but also uses techniques such as Neural Networks which can generate valid predictions but are not capable of identifying the specific nature of the interrelations between the variables on which the predictions are based.

2.3 Introduction to Association Rule Mining [16]:

A number of data mining algorithms have been introduced to the community that perform summarization of the data, classification of data with respect to a target attribute, deviation detection and other forms of data characterization and interpretation. One popular summarization and pattern extraction algorithm is the association rule algorithm, which identifies correlations between items in transactional databases.

Given a set of transactions, each described by an unordered set of items, an association rule X → Y may be discovered in the data, where X and Y are conjunctions of items. The intuitive meaning of such a rule is that, transactions in the database, which contain the items in X, also tend to contain the items in Y.

An example of such a rule might be that many observed customers who purchase tires and auto accessories also buy some automotive services. In this case, X = {tires, auto accessories} and Y = {automotive services}. Two numbers are associated with each rule that indicates the support and confidence of the rule.

The support of the rule X → Y represents the percentage of transactions from the original database that contain both X and Y.

The confidence of rule X → Y represents the percentage of transactions containing items in X that also contain items in Y.

Applications of association rule mining include cross marketing, attached mailing, catalog design and customer segmentation. An association rule discovery algorithm searches the space of all possible patterns for rules that meet the user-specified support and confidence thresholds.

2.3.1 Apriori Algorithm:

Apriori algorithm is an example of association rule mining algorithm. Using this algorithm, candidate patterns that receive sufficient support from the database are considered for transformation into a rule. This type of algorithm works well for complete data with discrete values.

One limitation of many association rule mining algorithms such as the Apriori algorithm is that only database entries, which exactly match the candidate patterns, may contribute to the support of the candidate pattern. This creates a problem for databases containing many small variations between otherwise similar patterns and for databases containing missing values.

The problem of discovering association rules can be divided into two steps:

- Find all item sets whose support is greater than the specified threshold. Item sets with minimum support are called frequent item sets.
- Generate association rules from the frequent item sets. To do this, consider all partitioning of the item set into rule's left-hand and right-hand sides. Confidence of a candidate rule X→ Y is calculated as support (XY) / support (X). All rules that meet the confidence threshold are reported as discoveries of the algorithm.

Join Step : C_k is generated by joining Lk-1with itself
Prune Step : Any (k-1)-itemset that is not frequent cannot be a subset of a frequent k-itemset

Pseudo-code:

C k : Candidate itemset of size k
Lk : frequent itemset of size k

```
• L1= {frequent items};
• for(k= 1; Lk!=∅; k++) do begin
• Ck+1= candidates generated from Lk;
• for each transaction t in database do
• increment the count of all candidates in Ck+1that are contained in t
• Lk+1= candidates in Ck+1with min_support
• end
• return∪kLk;
```

Report UkLk as the discovered frequent itemsets

k-item set	An item set containing k items
Lk	Set of frequent k-item sets(k-itemsets with minimum support)
Ck	Set of candidate k-itemsets (potentially frequent itemsets)
UkLk	Set of generated itemsets

Table 2.1 Apriori Algorithm

Table 2.1 summarizes the Apriori algorithm. The first pass of the algorithm calculates single item frequencies to determine the frequent 1-itemsets. Each subsequent pass k, discovers frequent item sets of size k. To do this, the frequent item sets Lk-1 found in the previous iteration are joined to generate the candidate item sets Ck. Next, the support for candidates in Ck is calculated through one sweep of the transaction list.

From Lk-1, the set of all frequent (k-1) item sets - the set of candidate k-item sets is created. The intuition behind this Apriori candidate generation procedure is that if an item set X has minimum support, so do all the subsets of X. Thus new item sets are created from (k-1) itemsets p and q by listing p.item1, p.item2, p.item (k-1), q.item (k-1). Items p and q are selected if items 1 through k-2 are equivalent for p and q and item k-1 is not equivalent. Once candidates are generated, items are removed from consideration if any (k-1) subset of the candidate is not in Lk-1

2.4. Fuzzy Logic Concepts:

2.4.1. Fuzzy Logic:

Fuzzy logic starts with and builds on a set of user supplied human language rules. The fuzzy systems convert these rules to their mathematical equivalents. This simplifies the job of the system designer and the computer, and results in much more accurate representation of the way systems behave in the real world.

Additional benefits of fuzzy logic include its simplicity and its flexibility. Fuzzy logic can handle problems with imprecise and incomplete data, and it can model nonlinear functions of arbitrary complexity.

A fuzzy system can create to match any set of input data. The Fuzzy Logic Toolbox makes this particularly easy by supplying adaptive techniques such as adaptive neuro-fuzzy inference systems (ANFIS) and fuzzy subtractive clustering. Fuzzy logic models, called fuzzy inference systems, consist of a number of confidential "if then" rules.

In fuzzy logic, unlike standard conditional logic, the truth of any statement is a matter of degree. The inference rule is the form of p -- > q (p implies q). For example, the rule if (weather is cold) then (heat is on), both variables, cold and on, has ranges of values. Fuzzy inference systems rely on membership functions to explain to the computer how to calculate the correct value between 0 and 1. The degree to which any fuzzy statement is true is denoted by a value between 0 and 1. Not only do the rule-based approach and flexible membership function scheme make fuzzy systems straightforward to create, but they also simplify the design of systems and ensure that it can easily update and maintain the system over time.

2.4.2. Fuzzy Set [15]:

Bivalent Set Theory can be somewhat limiting if we wish to describe a 'humanistic' problem mathematically. For example, Fig. 2.1 illustrates bivalent sets to characterize the temperature of a room.

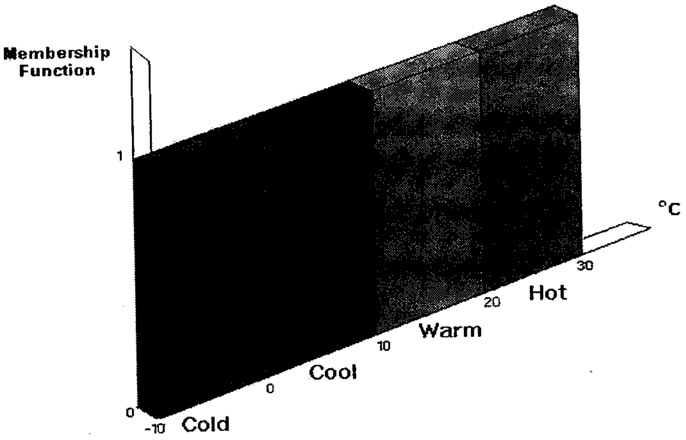


Fig. 2.1 Bivalent sets to characterize temperature of a room

The most obvious limiting feature of bivalent sets that can be seen clearly from the diagram is that they are mutually exclusive - it is not possible to have membership of more than one set (opinion would widely vary as to whether 50 degrees Fahrenheit is 'cold' or 'cool' hence the expert knowledge we need to define our system is mathematically at odds with the humanistic world). Clearly, it is not accurate to define a transition from a quantity such as 'warm' to 'hot' by the application of one degree Fahrenheit of heat. In the real world a smooth (unnoticeable) drift from warm to hot would occur. This natural phenomenon can be described more accurately by Fuzzy Set Theory. Fig. 2.2 shows how fuzzy sets quantifying the same information can describe this natural drift.

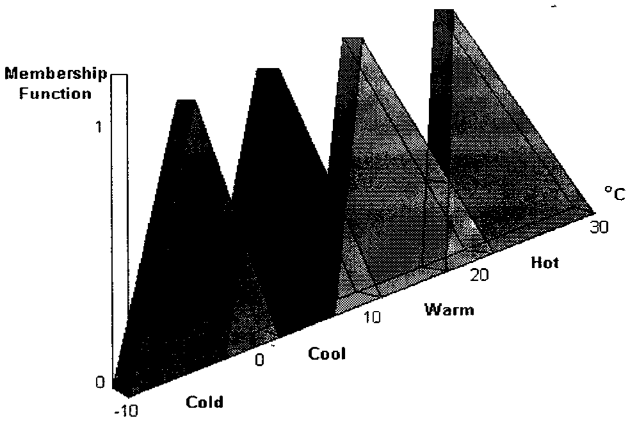


Fig. 2.2 Fuzzy sets to characterize temperature of a room

The whole concept can be illustrated with this example. Let's talk about people and "youthness". In this case the set S (the universe of discourse) is the set of people. A fuzzy subset YOUNG is also defined, which answers the question "to what degree is person x young?" To each person in the universe of discourse, we have to assign a degree of membership in the fuzzy subset YOUNG. The easiest way to do this is with a membership function based on the person's age.

$$\text{young}(x) = \{\; \begin{array}{ll} 1, & \text{if age}(x) <= 20, \\ (30-\text{age}(x))/10, & \text{if } 20 < \text{age}(x) <= 30, \\ 0, & \text{if age}(x) > 30 \;\} \end{array}$$

A graph of this looks like the one in Fig. 2.3:

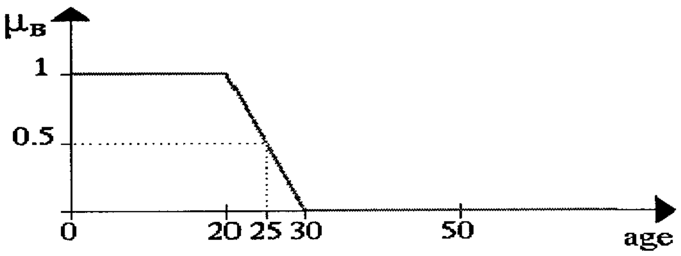


Fig. 2.3 Graph to Characterize Person's Age.

2.4.3 Fuzzy Set Operations[16] :

Universe Of Discourse :

The Universe of Discourse is the range of all values for an input to a fuzzy system.

Fuzzy Set :

A fuzzy Set is any set that allows its member to have different grades of membership in the interval.

Membership Function :

The membership function μA(x) quantifies the grade of membership of the elements x to the fundamental set X. An element mapping to the value 0 means that the member is not included in the given set, 1 describes a fully included member. Values strictly between 0 and 1 characterize the fuzzy members as shown in figure 2.4

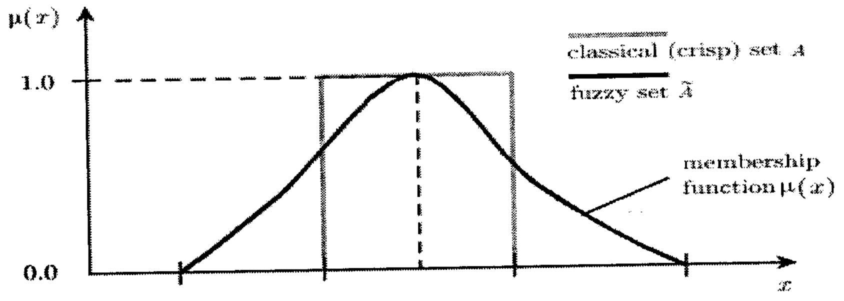


Fig. 2.4 Membership Function

Types of membership functions [18]:

1. Numerical definition (discrete membership functions)

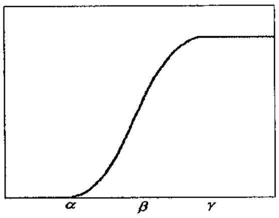
$$A=\sum_{x_i\in X}\mu_A(x_i)/x_i$$

2. Function definition (continuous membership functions)

Including of S function, Z Function, Pi function, Triangular shape, Trapezoid shape, Bell shape.

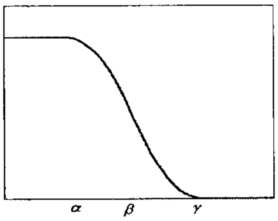
$$A = \int_X \mu_A(x)/x$$

(1) S **Function**: monotonical increasing membership function



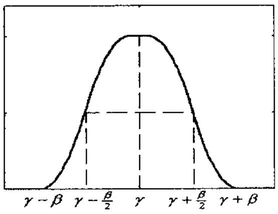
$$S(x;\alpha,\beta,\gamma)=\begin{cases} 0 & \textit{for } x\le\alpha \\ 2(\frac{x-\alpha}{\gamma-\alpha})^2 & \textit{for } \alpha\le x\le\beta \\ 1-2(\frac{x-\alpha}{\gamma-\alpha})^2 & \textit{for } \beta\le x\le\gamma \\ 1 & \textit{for } \gamma\le x \end{cases}$$

(2) **Z Function**: monotonical decreasing membership function



$$Z(x;\alpha,\beta,\gamma)=\begin{cases} 1 & \textit{for } x\le\alpha \\ 1-2(\frac{x-\alpha}{\gamma-\alpha})^2 & \textit{for } \alpha\le x\le\beta \\ 2(\frac{x-\alpha}{\gamma-\alpha})^2 & \textit{for } \beta\le x\le\gamma \\ 0 & \textit{for } \gamma\le x \end{cases}$$

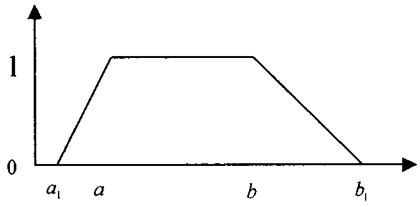
(3) Π **Function**: combine S function and Z function, monotonically increasing and decreasing membership function



$$\Pi(x;\beta,\gamma)=\begin{cases} S(x;\gamma-\beta,\gamma-\frac{\beta}{2},\gamma) & \textit{for } x\le\gamma \\ 1-S(x;\gamma,\gamma+\frac{\beta}{2},\gamma+\beta) & \textit{for } x\ge\gamma \end{cases}$$

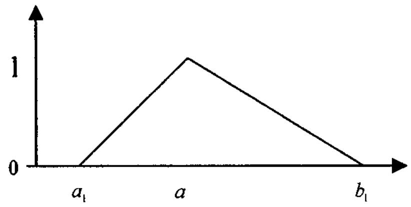
Piecewise continuous membership function

(4)Trapezoidal Membership Function



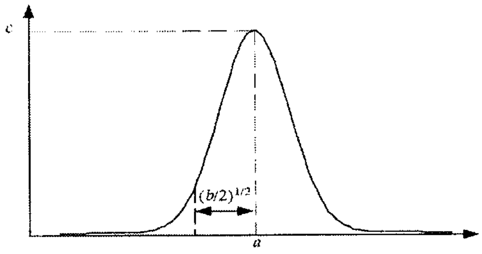
$$\mu_A(x) = \begin{cases} 0 & for\ x \le a_1 \\ \frac{x-a_1}{a-a_1} & for\ a_1 \le x \le a \\ 1 & for\ a \le x \le b \\ \frac{b_1-x}{b_1-b} & for\ b \le x \le b_1 \\ 0 & for\ b_1 \le x \end{cases}$$

(5) Triangular Membership Function



$$\mu_A(x) = \begin{cases} 0 & for\ x \le a_1 \\ \frac{x-a_1}{a-a_1} & for\ a_1 \le x \le a \\ \frac{b_1-x}{b_1-a} & for\ a \le x \le b_1 \\ 0 & for\ b_1 \le x \end{cases}$$

(6) Bell-shaped membership function



3. LITERATURE SURVEY

3.1 Reconstruction-Based Association Rule Hiding [2] :

The proposed framework first performs sanitization on an itemset lattice called a knowledge base from which association rules can be derived. The itemset lattice is defined as all partial ordered subset items generated from given transactions. Then a reconstruction procedure reconstructs a new released dataset from the sanitized itemset lattice. In one word, this approach conceals the sensitive rules by sanitizing itemset lattice rather than sanitizing original dataset. In this way, one can easily control the availability of rules that can be mined from original dataset and control the hiding effects directly. However, as a rudimental work, the approach is still very incomplete and limited in the following two aspects:

1) It does not give concrete guidance on how to sanitize the itemset lattice according to the sensitive association rules.

2) The feasibility of the data reconstruction process is restricted to whether the knowledge sanitization process can produce an itemset lattice with consistent support value configuration relationship.

3.2 Mining Quantitative Association Rules in Large Relational Tables [5] :

In this paper, the problem of mining association rules over quantitative and categorical attributes in large relational tables and techniques for discovering such rules are presented. The range of value of quantitative attribute is divided into intervals. Unfortunately, increasing the number of intervals while simultaneously combining adjacent intervals introduces two new problems:

ExecTime: If a quantitative attribute has n values (or intervals), there are on average O(n) ranges that include a specific value or interval. Hence the number of items per record blows up, which will blow up the execution time

ManyRules: If a value (or interval) of a quantitative attribute has minimum support, so will any range containing this value/interval. Thus, the number of rules blows up. Many of these rules will not be interesting.

To mitigate the "Exec Time" problem, the extent to which adjacent values/intervals may be combined is restricted by introducing a user-specified "maximum support" parameter; if the combined support of the intervals exceeds this value, combining the intervals is stopped. To address the "ManyRules" problem, an interest measure is used. The interest measure is based on deviation from expectation and helps prune out uninteresting rules. This algorithm shares the basic structure of the algorithm for finding Boolean association rules.

3.3 A Reconstruction-based Algorithm for Classification Rules Hiding [4] :

In this paper, the focus is on the classification rules hiding. The classification rules hiding framework was proposed for categorical datasets by using reconstruction technique. Instead of arbitrary dataset modification, the framework reconstructs a new dataset that only non-sensitive rules can be discovered from it. Additionally, the usability of the new dataset is also addressed. The method is as follows: a rule-based classification algorithm is used on the given dataset to obtain all classification rules. Subsequently, only non-sensitive classification rules which are approved by the data owner are used to build a data generator, a decision tree. Finally, a new dataset which contains only non-sensitive classification rules is reconstructed from the decision tree.

3.4 Algorithms for Balancing Privacy and Knowledge Discovery in Association Rule Mining [6] :

The procedure of converting an original database into a sanitized one is called the sanitization process. To do so, a small number of transactions have to be modified by deleting one or more items from them or even adding noise to the data by turning some items from 0 to 1 in some transactions. This approach relies on boolean association rules. On the other hand, such an approach must hold the following restrictions:

- The impact on the non-restricted data has to be minimal
- An appropriate balance between a need for privacy and knowledge discovery must be guaranteed.

To accomplish these restrictions, Sanitizing algorithms require only two scans regardless of the database size and the number of restrictive association rules that must be protected. The first scan is required to build the index for speeding up the sanitization process, while the second scan is used to sanitize the original database. This represents a significant improvement over the previous algorithms presented in the literature, which require various scans depending on the number of association rules to be hidden.

3.5 Using Unknowns to Prevent Discovery of Association Rules [10] :

The technique presented here applies to applications where it is necessary to store imprecise or unknown values for some attributes, such as when actual values are confidential or not available. An innovative technique for hiding rules from a data set, by replacing select attribute values with unknowns is proposed. Sometimes false values can have bad consequences. Therefore, for many situations it is safer if the sanitization process place unknown values instead of false values. The goal of the algorithms is to obscure a given set of sensitive rules by replacing known values with unknowns, while minimizing the side effects on non-sensitive rules.

3.6 Privacy Preserving Frequent Itemset Mining [1] :

Major novelty with this approach is that it take into account the impact of sanitization not only on hiding the patterns that should be hidden but also on hiding legitimate patterns that should not be hidden. Thus, this framework tries to find a balance between privacy and disclosure of information by attempting to minimize the impact on the sanitized transactions. Other approaches presented in the literature focus on the hiding of restrictive patterns but do not study the efficient of their sanitization on accidentally concealing legitimate patterns or even generating artifact patterns.

4. OBJECTIVE

The cojective of the system is

- To mine fuzzy association rules from the quantitative data.
- To provide input privacy and output privacy in the mining operations
- The critical rules are determined which are used to provide input privacy and the data is altered to output privacy
- To prove that the system produces consistent result
- The system is designed to secure the patient data with privacy preservation mechanisms. The breast cancer data set is used for the system

5. SYSTEM METHODOLOGY

Knowledge Discovery in Databases (KDD) means the application of non-trivial procedures for identifying effective, coherent, potentially useful, and previously unknown patterns in large databases. The KDD process generally consists of the following three phases [11].

(1) Pre-processing: This consists of all the actions taken before the actual data analysis process starts. It may be performed on the data for the following reasons: solving data problems that may prevent us from performing any type of analysis on the data, understanding the nature of the data, performing a more meaningful data analysis, and extracting more meaningful knowledge from a given set of data.

(2) Data-mining: This involves applying specific algorithms for extracting patterns or rules from data sets in a particular representation.

(3) Post-processing: This translates discovered patterns into forms acceptable for human beings. It may also make possible visualization of extracted patterns.

Data-mining is most commonly used in attempts to induce association rules from transaction data. Most previous studies have only shown, however, how binary valued transaction data may be handled. Transaction data in real-world applications do not usually consist of quantitative values, so designing a sophisticated data-mining algorithm and being able to deal with various types of data presents a challenge to workers in this research field.

Fuzzy set theory is being used more and more frequently in intelligent systems because of its simplicity and similarity to human reasoning. The theory has been applied in fields such as manufacturing, engineering, diagnosis, economics, among others.

This project integrates fuzzy-set concepts with the apriori mining algorithm [4] and uses the result to find interesting itemsets and fuzzy association rules in transaction data with quantitative values.

A new mining algorithm, called the fuzzy transaction data-mining algorithm (FTDA) is proposed[12]. It transforms quantitative values in transactions into linguistic terms, then alters them to find association rules by modifying the apriori mining algorithm. Then it uses the Increase the Support of Left hand side(ISL) method to hide critical association rules.

The system is divided into four major modules

- Dataset preparation
- Fuzzification Process
- Rule mining and hiding process
- Experimental Results

5.1 Dataset Preparation:

This module involves pre – processing of the dataset. The algorithm was applied to the Wisconsin Breast Cancer database from University of California, Irvine(UCI) Machine Learning Repository[17]. The database consists of 10 attributes out of which one is categorical. The number of instances in the dataset is 699. There are 16 instances in dataset that contain a single missing (i.e., unavailable) attribute value, denoted by "?". These missing value are filled by finding the mean of the particular attribute. There are 54 instances that are redundant. Redundant records denote repeated diagnosis and this redundancy was removed by deleting the old diagnostic values. Provisions are provided for the users to add and delete records.

- The attributes are

#	Attribute	Domain
1.	Sample code number	id number
2.	Clump Thickness	1 - 10
3.	Uniformity of Cell Size	1 - 10
4.	Uniformity of Cell Shape	1- 10

5. Marginal Adhesion	1 - 10
6. Single Epithelial Cell Size	1 - 10
7. Bare Nuclei	1 - 10
8. Bland Chromatin	1 - 10
9. Normal Nucleoli	1 - 10
10. Mitoses	1 - 10
11. Class: (2 for benign, 4 for malignant)	

5.2 Fuzzification Process:

A *fuzzy set* is a set without a crisp, clearly defined boundary. It can contain elements with only a partial degree of membership. The data values are mapped into fuzzy values using triangular membership functions. The domain of the attribute values are distributed into three fuzzy regions as shown in the figure.5.1

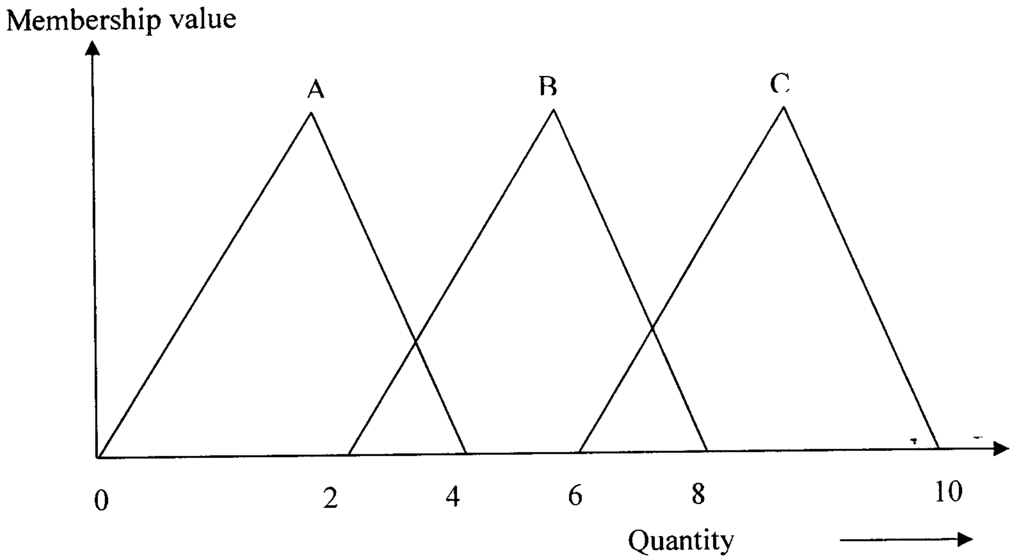


Fig. 5.1 Triangular membership function to map data values to fuzzy values

5.3 Rule mining and hiding process:

The association rules are mined using the Fuzzy Transaction Data Mining (FTDA) algorithm [11] as follows:

5.3.1 The Proposed Algorithm:

In a quantitative database, if a critical rule X→Y needs to be hidden, its confidence value is decreased to a value smaller than the minimum confidence value. One way of decreasing confidence value is increasing the support value of an item X at LHS (Left Hand Side), and the other way is decreasing support value of an item Y at RHS (Right Hand Side) to a value lower than the minimum support value.

In this method, in order to decrease confidence value of a rule, the support value of an item at LHS is increased. For this purpose, the value of item in LHS is subtracted from 1 in the case the value of item in LHS is lower than 0.5 and than value of item in RHS.Abbreviations used in the proposed algorithm are given as follows:

Initial database with n transaction data, D; fuzzified database, F; a set of predicting items, X; transactions belong to a LHS item, TL; transactions belong to a RHS item, TR; rule, U.

Input:-

- A source database D,
- A min_support
- A min_confidence
- A set of predicting items X

5.3.2 Stages of the Algorithm[8]:

An association rule is defined as an implication X→Y, where both X and Y are defined as sets of attributes (interchangeably called items) . Here X is called as the body or Left Hand Side(LHS) of the rule and Y is called as the head or Right Hand Side(RHS) of the rule; it is interpreted as follows: "for a specified fraction of the existing transactions, a particular value of an attribute set X determines the value of attribute set Y as another particular value under a certain confidence". For instance, an association rule in a supermarket basket data may be stated as, "in 20% of the transactions, 75% of the people buying butter also buy milk in the same transaction"; 20% and 75% represent the support and the confidence, respectively. The significance of an association rule is measured by its support and confidence. Simply, support is the percentage of transactions that contain both X and Y, while confidence is the ratio of the support of Y X U to the support of X.

So, the problem can be stated as:

- Find all association rules that satisfy user-specified minimum support and confidence.

STEP 1: Transform the quantitative value v_j of each item i_j entered into a fuzzy set represented as $(\frac{F_{j1}}{R_{j1}}+\frac{F_{j2}}{R_{j2}}+ \ldots\ldots \frac{F_{jl}}{R_{jl}})$ by using the given membership functions for item quantities, where l is the number of regions for i_j.

STEP 2: Calculate the count of each attribute region (linguistic term) R_{jk} in the transaction data as:

$$\text{Count}_{jk} = \sum_{i=1}^{n} f_{jk}$$

STEP 3: Check whether count_{jk} of each R_{jk} over n is larger than or equal to the predefined minimum support value. If R_{jk} satisfies the above condition, put it in the set of large-1 itemsets (L_1).

STEP 4: Join the large itemsets L_1 to generate the candidate set C_2. Two regions belonging to the same item cannot simultaneously exist in an itemset in C_2.

STEP 5 : Calculate the fuzzy value of each transaction data as:

$$f_I = \overset{2}{\underset{j=1}{Min}} f_{Ij}$$

Then, calculate the fuzzy count of I in the transactions as:

$$Count_I = \sum_{i=1}^{n} f_I^i$$

STEP 6 : According to user specified minimum confidence value, rules are extracted. A confidence value of a Ao→ Bo rule is computed as follows:

$$\text{Confidence (A0→B0)} = \frac{Support(AoBo)}{Support(Ao)}$$

STEP 7 : Critical rules are determined. Then in order to hide the rules, confidence values are tried to be decreased. This is achieved by one of two strategies. The first one is to increase the support count of Ao, i.e., LHS of the rule, but not support count of Ao . The second one is to decrease the support count of the itemset $Ao \cup Bo$

$$\text{Confidence (A0→B0)} = \frac{Support(AoBo)\downarrow}{Support(Ao)\uparrow}$$

6. RESULTS AND CONCLUSION

6.1 Experimental Results:

In order to understand the characteristics of the algorithm in a numerical way, several experiments were done and the output effects were observed. For the experiments, a computer having Intel Pentium 4 processor and 512MB RAM was used. The algorithm was applied to the Wisconsin Breast Cancer database from UCI Machine Learning Repository [17]. The database consists of 10 attributes out of which one is categorical.

Three different experiments were conducted.

The first is to show the relationship between number of total and hidden rules, and number of transactions. In this experiment, the minimum support values taken are 5.0, 5.0, 8.0 and 15.0 respectively and minimum confidence value is set at 10%. The results are depicted in Fig. 6.1.

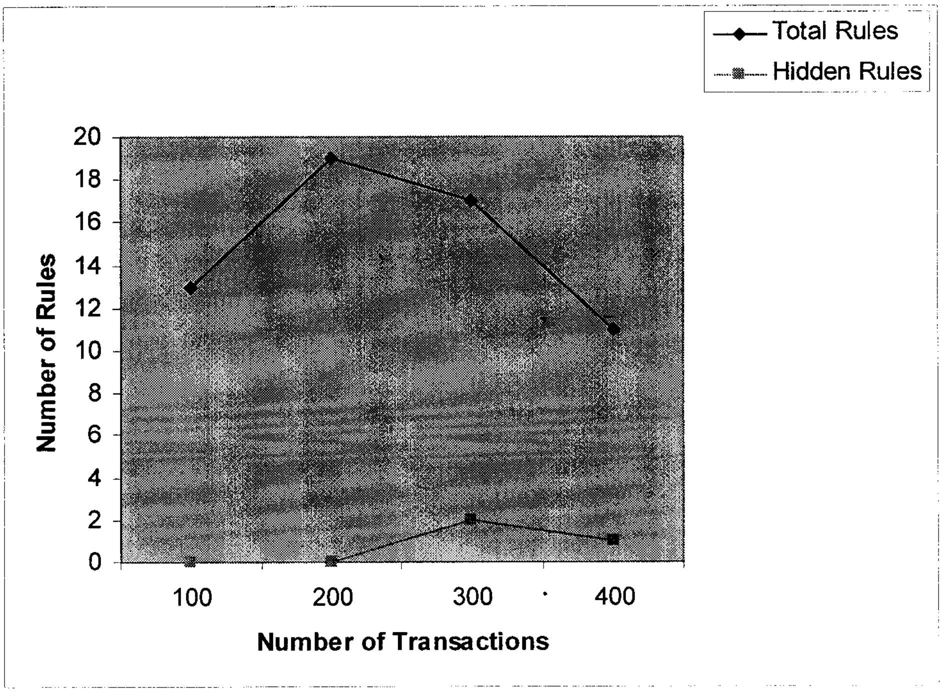


Fig.6.1. Number of total and hidden rules

The second experiment deals with finding the number of total and hidden rules for different values of minimum support. As can be seen from fig. 6.2 the number of rule decreases with increase of minimum support value. In this experiment, the size of data set is 645.

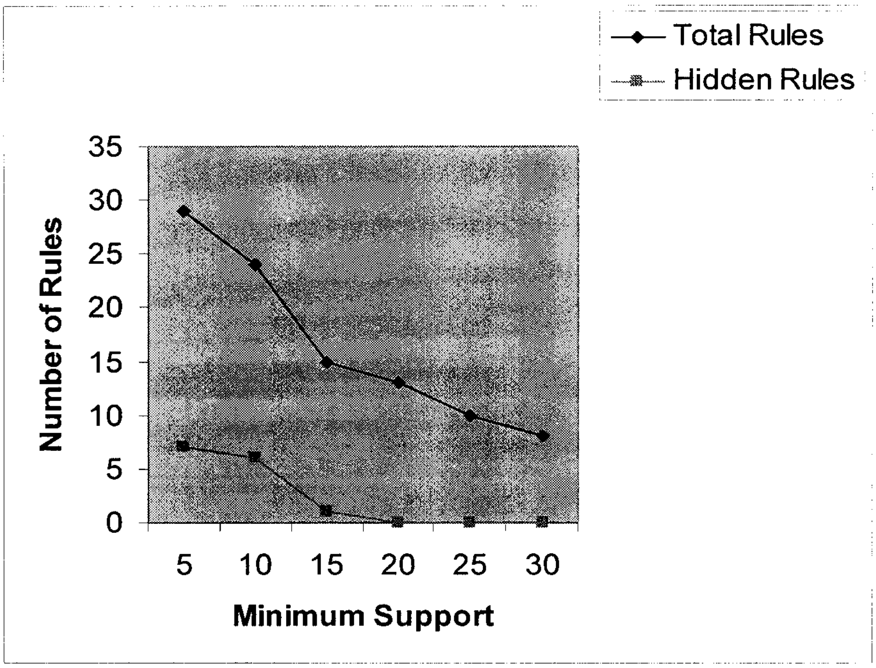


Fig. 6.2. The number of rules under different minimum support values

The third experiment finds the number of total and hidden rules for different values of minimum confidence. The results are reported in fig. 6.3 which demonstrates that the number of hidden rules quickly rises with the increase of minimum confidence value as the number of total rule decreases slowly.

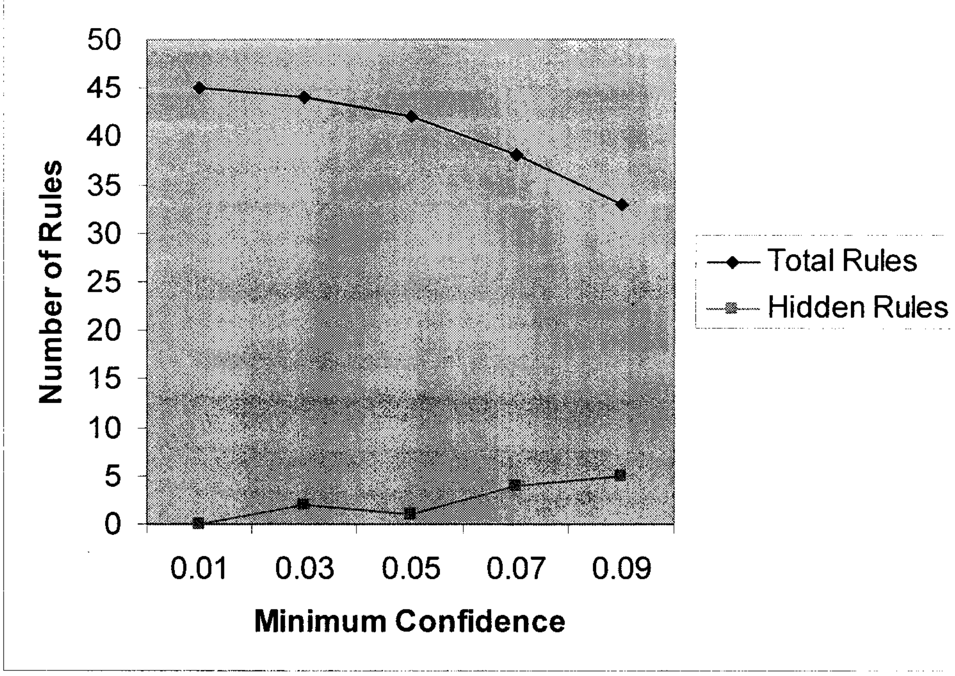


Fig. 6.3 The number of rules under different minimum confidence values

6.2 Conclusion:

In this project work, a privacy preserving data mining method by hiding fuzzy association rules was proposed. Unlike classical approaches, this method handles hiding the association rules in quantitative datasets. For this purpose, it employs fuzzy concepts. Experiments conducted on the Breast Cancer dataset illustrated that the proposed approach produces meaningful results and has reasonable efficiency. The results of the proposed algorithm are consistent and hence encouraging.

6.3 Future Scope:

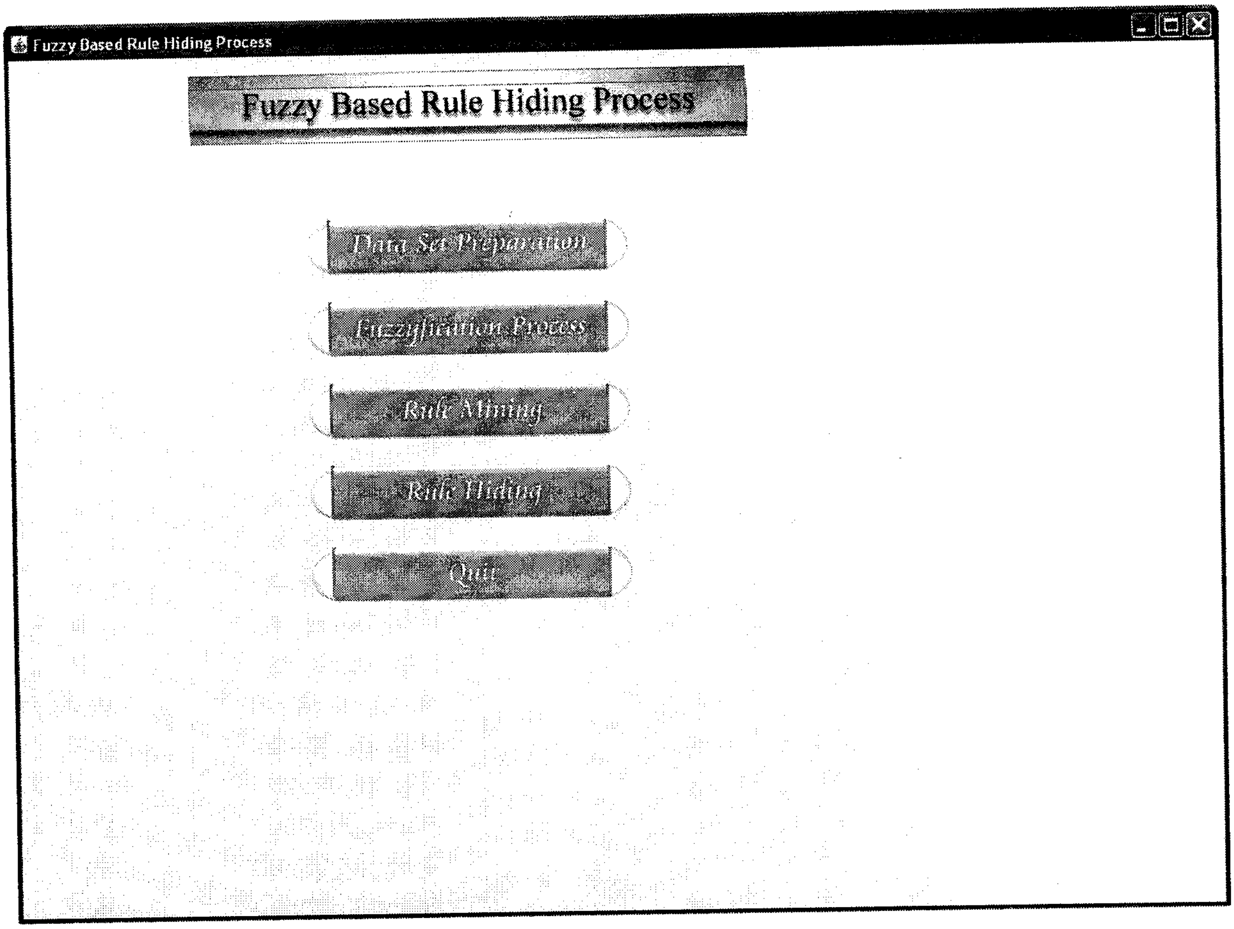
In existing approaches, fuzzy sets are either supplied by an expert or determined by applying an existing clustering algorithm. The former is not realistic, because it is extremely hard for an expert to specify fuzzy sets. The latter approaches have not produced satisfactory results. They have not considered the optimization of membership functions. A user specifies the number of fuzzy sets and membership functions are tuned accordingly.

So, future extension is to find a clustering method that employs multi objective genetic algorithm for the automatic discovery of membership functions used in determining fuzzy quantitative association rules. This method has to optimize the number of fuzzy sets and their ranges according to multi-objective criteria in a way to maximize the number of large item sets with respect to a given minimum support value.

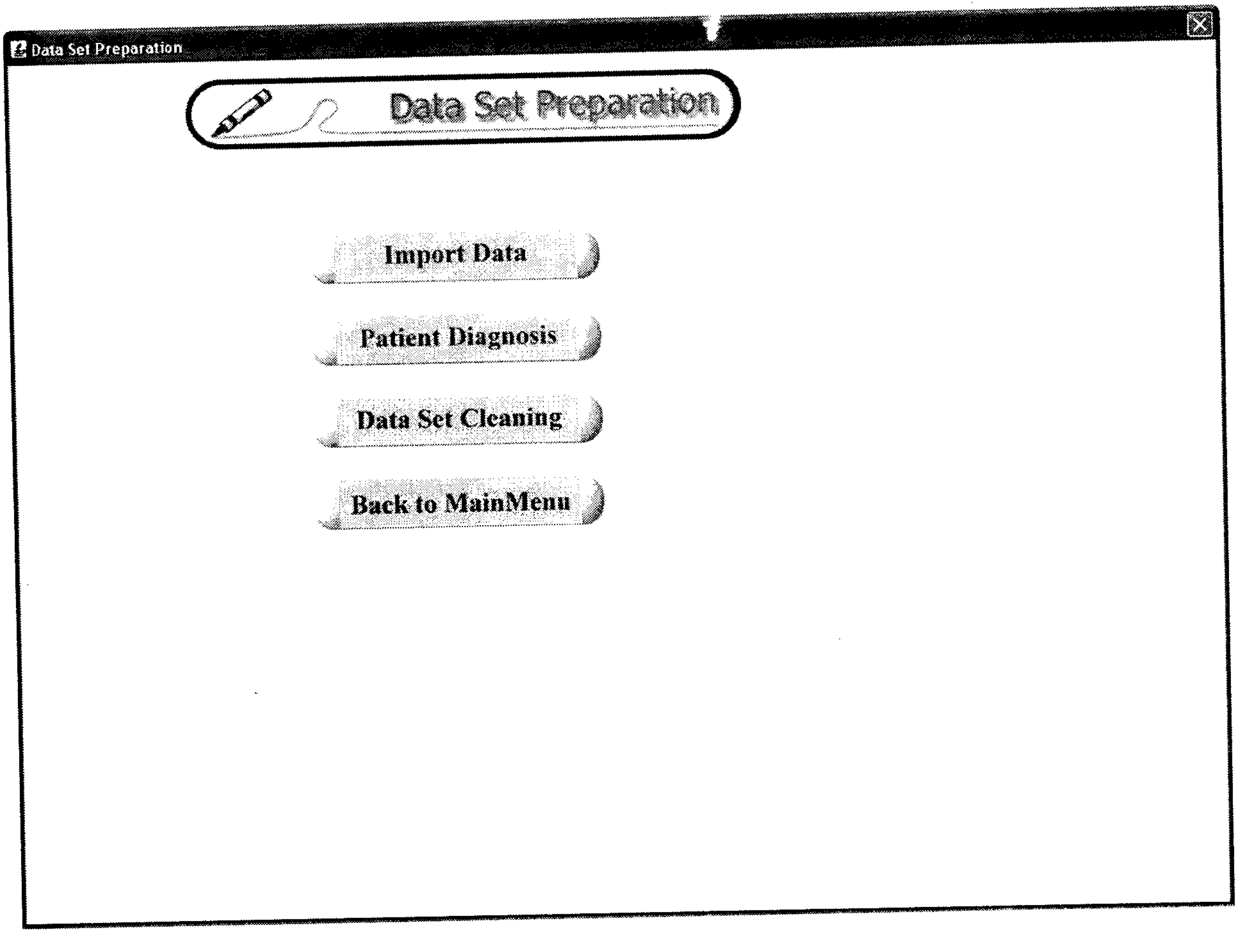
APPENDIX - 1

Screenshots

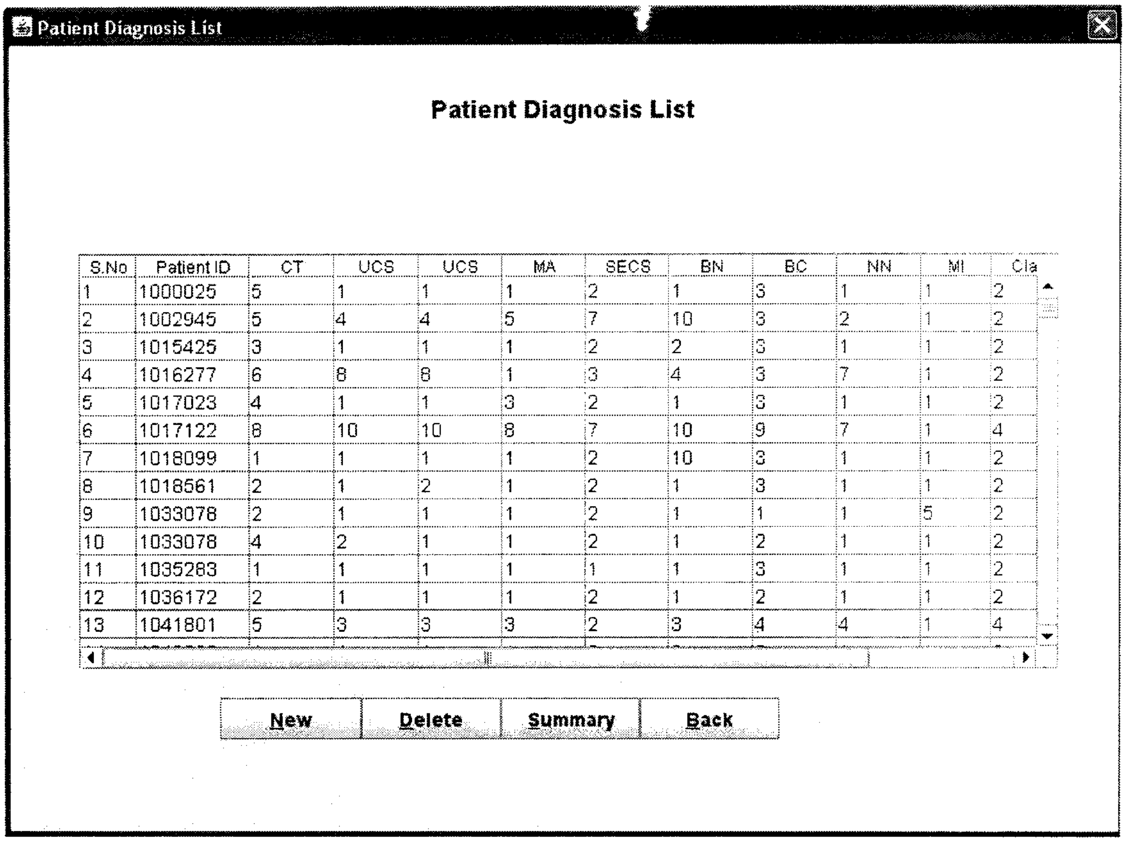
Main Menu



Dataset Preparation Menu



Patient Diagnosis List

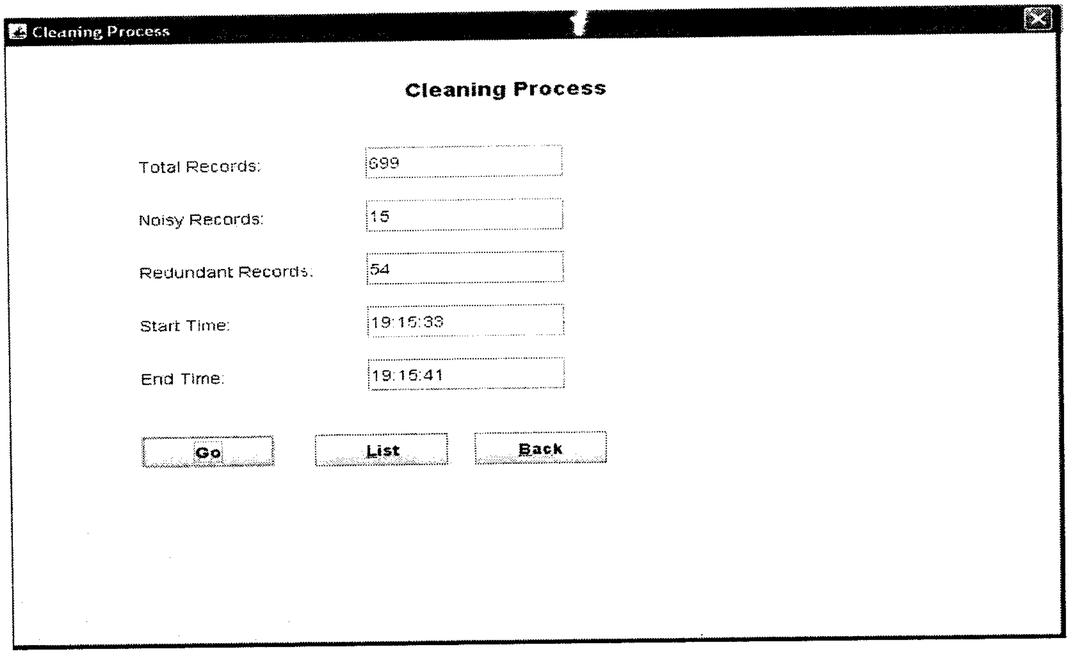


Patient Diagnosis List

S.No	Patient ID	CT	UCS	UCS	MA	SECS	BN	BC	NN	MI	Cla
1	1000025	5	1	1	1	2	1	3	1	1	2
2	1002945	5	4	4	5	7	10	3	2	1	2
3	1015425	3	1	1	1	2	2	3	1	1	2
4	1016277	6	8	8	1	3	4	3	7	1	2
5	1017023	4	1	1	3	2	1	3	1	1	2
6	1017122	8	10	10	8	7	10	9	7	1	4
7	1018099	1	1	1	1	2	10	3	1	1	2
8	1018561	2	1	2	1	2	1	3	1	1	2
9	1033078	2	1	1	1	2	1	1	1	5	2
10	1033078	4	2	1	1	2	1	2	1	1	2
11	1035283	1	1	1	1	1	1	3	1	1	2
12	1036172	2	1	1	1	2	1	2	1	1	2
13	1041801	5	3	3	3	2	3	4	4	1	4

This form provides option to add new patient detail and delete existing record. It also provides summary information about the number of records present for a given patient ID.

Cleaning Process



Cleaning Process

Total Records: 699

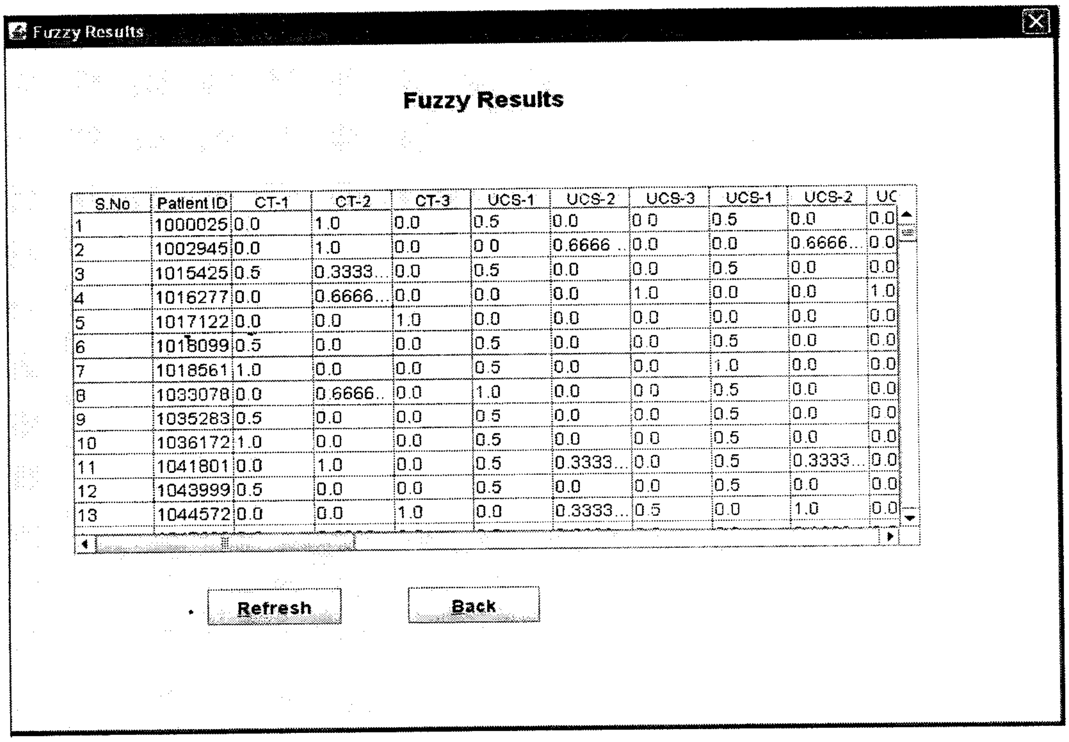
Noisy Records: 15

Redundant Records: 54

Start Time: 19:15:33

End Time: 19:15:41

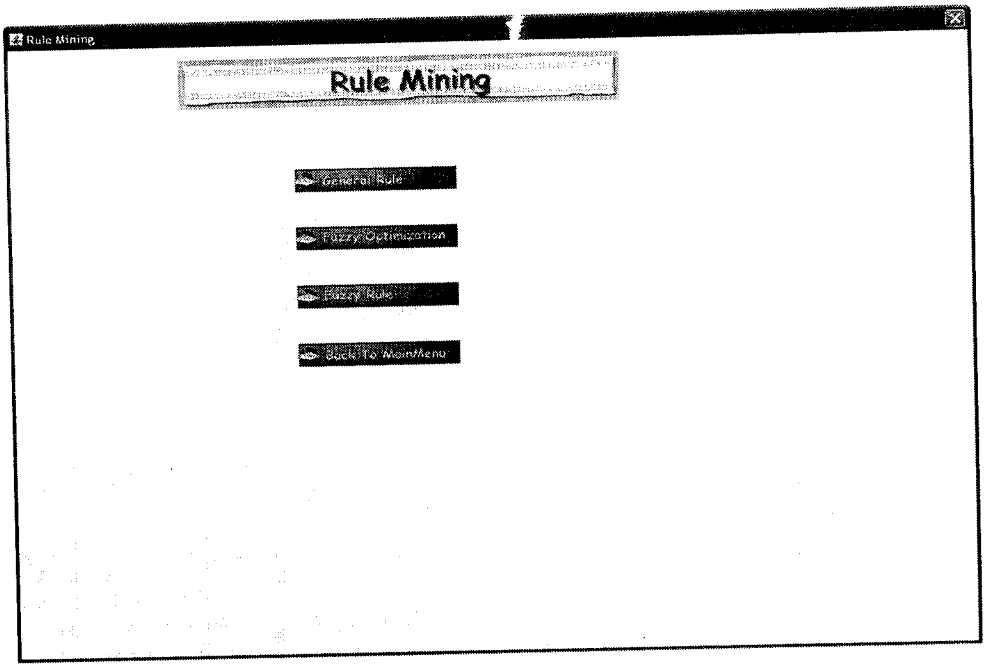
Dataset after fuzzification



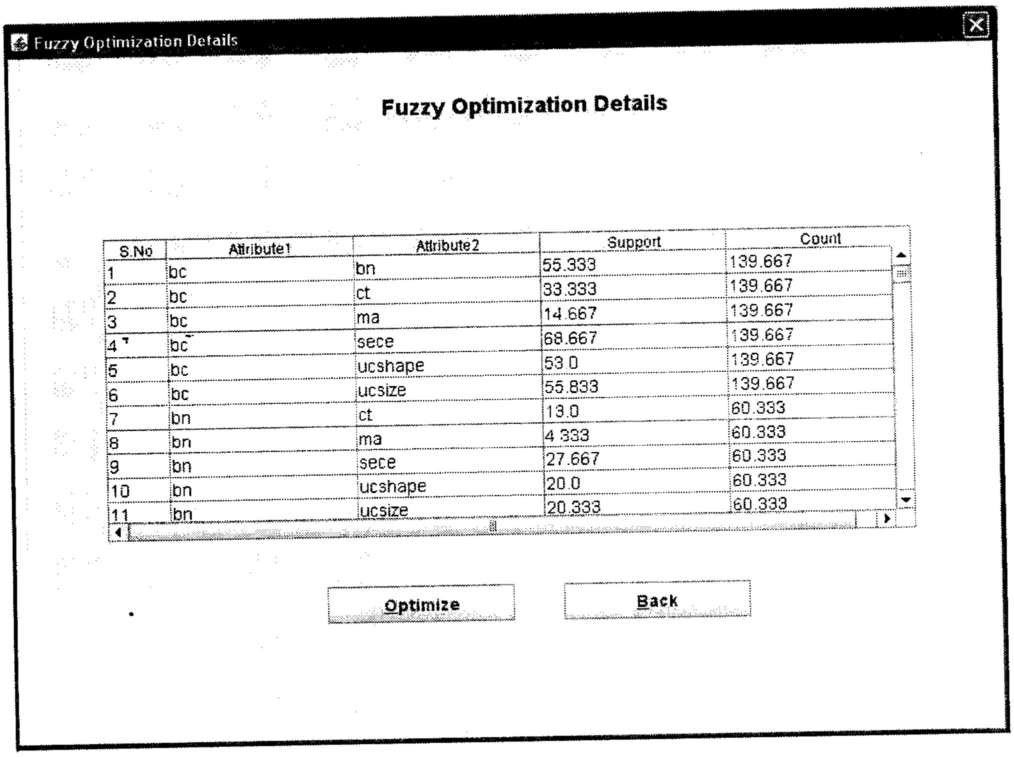
Fuzzy Results

S.No	Patient ID	CT-1	CT-2	CT-3	UCS-1	UCS-2	UCS-3	UCS-1	UCS-2	UC
1	1000025	0.0	1.0	0.0	0.5	0.0	0.0	0.5	0.0	0.0
2	1002945	0.0	1.0	0.0	0.0	0.6666 ..	0.0	0.0	0.6666...	0.0
3	1015425	0.5	0.3333...	0.0	0.5	0.0	0.0	0.5	0.0	0.0
4	1016277	0.0	0.6666...	0.0	0.0	0.0	1.0	0.0	0.0	1.0
5	1017122	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
6	1018099	0.5	0.0	0.0	0.5	0.0	0.0	0.5	0.0	0.0
7	1018561	1.0	0.0	0.0	0.5	0.0	0.0	1.0	0.0	0.0
8	1033078	0.0	0.6666..	0.0	1.0	0.0	0.0	0.5	0.0	0.0
9	1035283	0.5	0.0	0.0	0.5	0.0	0.0	0.5	0.0	0.0
10	1036172	1.0	0.0	0.0	0.5	0.0	0.0	0.5	0.0	0.0
11	1041801	0.0	1.0	0.0	0.5	0.3333...	0.0	0.5	0.3333...	0.0
12	1043999	0.5	0.0	0.0	0.5	0.0	0.0	0.5	0.0	0.0
13	1044572	0.0	0.0	1.0	0.0	0.3333...	0.5	0.0	1.0	0.0

Menu for mining fuzzy rules



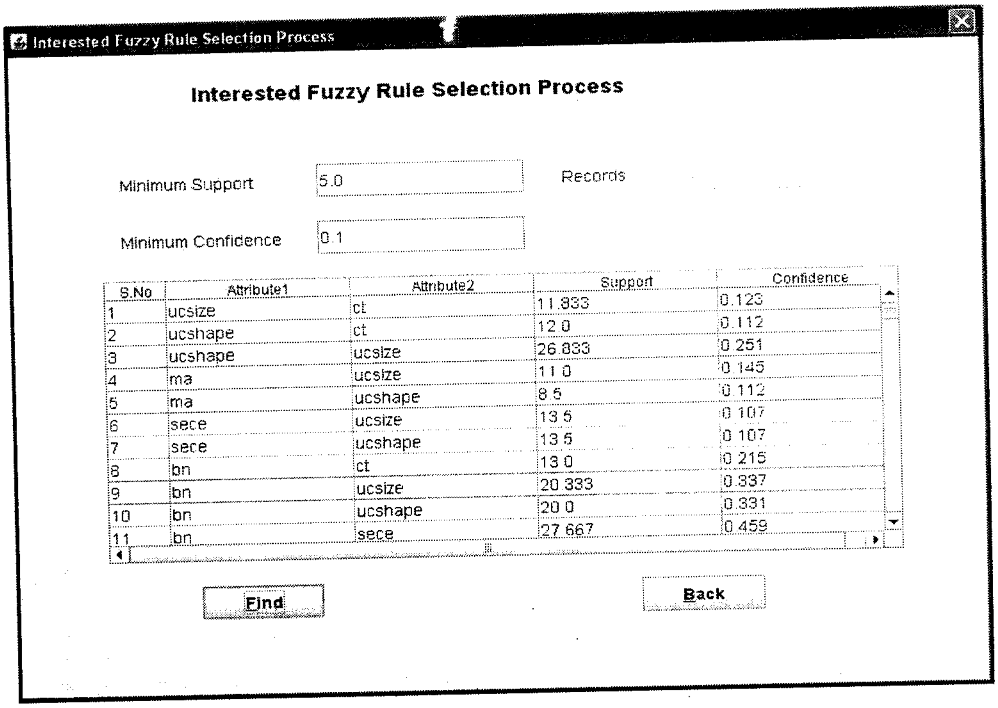
Finds support and count for the association rules



Fuzzy Optimization Details

S.No	Attribute1	Attribute2	Support	Count
1	bc	bn	55.333	139.667
2	bc	ct	33.333	139.667
3	bc	ma	14.667	139.667
4	bc	sece	68.667	139.667
5	bc	ucshape	53.0	139.667
6	bc	ucsize	55.833	139.667
7	bn	ct	13.0	60.333
8	bn	ma	4.333	60.333
9	bn	sece	27.667	60.333
10	bn	ucshape	20.0	60.333
11	bn	ucsize	20.333	60.333

Interesting fuzzy rules



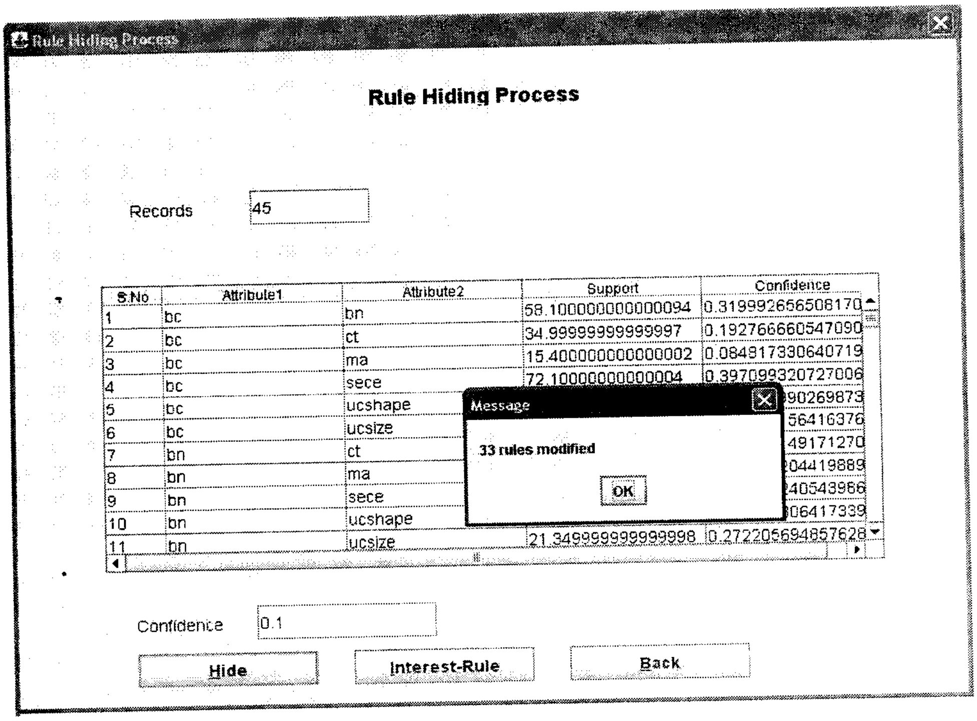
Interested Fuzzy Rule Selection Process

Minimum Support 5.0 Records

Minimum Confidence 0.1

S.No	Attribute1	Attribute2	Support	Confidence
1	ucsize	ct	11.833	0.123
2	ucshape	ct	12.0	0.112
3	ucshape	ucsize	26.833	0.251
4	ma	ucsize	11.0	0.145
5	ma	ucshape	8.5	0.112
6	sece	ucsize	13.5	0.107
7	sece	ucshape	13.5	0.107
8	bn	ct	13.0	0.215
9	bn	ucsize	20.333	0.337
10	bn	ucshape	20.0	0.331
11	bn	sece	27.667	0.459

Hiding those rules having confidence of 0.1



Rule Hiding Process

Records 45

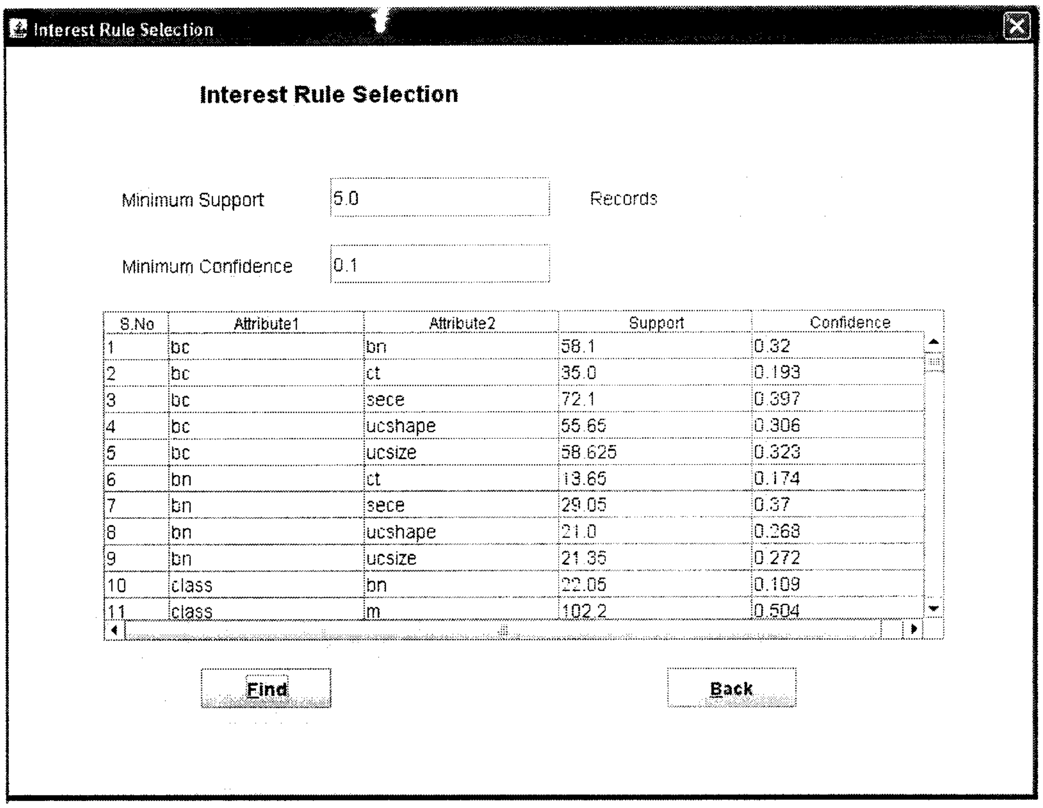
S.No	Attribute1	Attribute2	Support	Confidence
1	bc	bn	58.100000000000094	0.319992656508170
2	bc	ct	34.99999999999997	0.192766660547090
3	bc	ma	15.400000000000002	0.084817330640719
4	bc	sece	72.10000000000004	0.397099320727006
5	bc	ucshape	[illegible]	90269873
6	bc	ucsize	[illegible]	56416376
7	bn	ct	[illegible]	49171270
8	bn	ma	[illegible]	04419889
9	bn	sece	[illegible]	40543966
10	bn	ucshape	[illegible]	06417339
11	bn	ucsize	21.349999999999998	0.272205694857628

Message

33 rules modified

Confidence 0.1

Interesting rules after hiding those rules having confidence of 0.1



Interest Rule Selection

Minimum Support 5.0 Records

Minimum Confidence 0.1

S.No	Attribute1	Attribute2	Support	Confidence
1	bc	bn	58.1	0.32
2	bc	ct	35.0	0.193
3	bc	sece	72.1	0.397
4	bc	ucshape	55.65	0.306
5	bc	ucsize	58.625	0.323
6	bn	ct	13.65	0.174
7	bn	sece	29.05	0.37
8	bn	ucshape	21.0	0.268
9	bn	ucsize	21.35	0.272
10	class	bn	22.05	0.109
11	class	m	102.2	0.504

APPENDIX - 2

Sample Code

Clean process:

```
import java.io.*;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.table.*;

class ImportData extends JDialog implements ActionListener
{
        JLabel lbldatafilename,lblsize,lbllmodified,lbltitle;
        JTextField txtdatafilename,txtsize,txtlmodified;
        JPanel panel;
         JScrollPane jsp;
        JTextArea ta;
        WindowListener wlist;
        Font afont,tfont,ofont;
         private FileChooser fchooser;
         JButton btbrowse,btback,btimport,bttransfer;
         String fname = " ";
        DataMenu datamenu;
        String datapath = "", diagdata = "", lmdate = "";
         long datasize = 0;

        ImportData(DataMenu datamenu)
        {
                super(datamenu,"Import Diagnosis Data",true);
                this.datamenu = datamenu;
                  panel =(JPanel)getContentPane();
                  panel.setLayout(null);

                 int h =
                  ScrollPaneConstants.HORIZONTAL_SCROLLBAR_AS_NEEDED;
                   int v = ScrollPaneConstants.VERTICAL_SCROLLBAR_ALWAYS;

                lbltitle = new JLabel("<html><h2>Import Diagnosis Data</html></h2>");
                 lbltitle.setSize(300,30);
                 lbltitle.setLocation(300,30);
                 panel.add(lbltitle);

                 lbldatafilename = new JLabel("Data File Name:");
                lbldatafilename.setSize(150,30);
                lbldatafilename.setLocation(100,100);
                 lbldatafilename.setFont(tfont);
                 panel.add(lbldatafilename);

                btbrowse = new JButton("Browse");
                btbrowse.setSize(100,30);
                btbrowse.setMnemonic('B');

                wlist = new WindowAdapter()
                {
                public void windowClosing(WindowEvent we)
                {
                        dispose();
                 }
                 }
         addWindowListener(wlist);
          setSize(800,600);
        setLocation(0,0);
        setVisible(true);
        }
}
public void actionPerformed(ActionEvent ae)
{
        if(ae.getSource() == btback)
        {
                dispose();
        }
        if(ae.getSource() == btbrowse)
        {
                txtlmodified.setText("");
                 fchooser = new FileChooser();
                 fname = fchooser.getFileName();
                 if(fname == null)
                 {
                         Screen.showMessage("Select the File:");
                         return;
                 }
                 txtdatafilename.setText(fname);
                 datapath = fname;
                 File file = new File(datapath);
                 if(!file.exists())
                 {
                         Screen.showMessage("File not found");
                          return;
                 }
                datasize = file.length();
                long ldate = file.lastModified();
                lmdate = Tools.getDate(ldate) + " " + Tools.getTime(ldate);
                txtsize.setText(""+datasize+" Bytes");
                txtlmodified.setText(lmdate);
         }
        if(ae.getSource() == btimport)
        {
                 if(datapath.trim().length() == 0)
                {
                        Screen.showMessage("Select a data file");
                        return;
                 }
        assignData();
         }
      if(ae.getSource() == bttransfer)
       {
                 if(ta.getText().trim().length() == 0)
                {
                        Screen.showMessage("Data set not imported");
                         return;
                }
                DBTools.clearTable("rulehide","sdiagnosis");
                TransferProcess tp = new TransferProcess(datapath);
                tp.process();
                DBTools.updateRecordType();
                Screen.showMessage("TranferProcess Completed :");
        }
}
public void assignData()
{
        String data = "";
         diagdata = "";
         try
        {
                FileInputStream fis = new FileInputStream(datapath);
                 DataInputStream dis = new DataInputStream(fis);
                 while((data = dis.readLine()) != null)
                diagdata = diagdata + data + "\n";
                 dis.close();
                fis.close();
                ta.setText(diagdata);
        }catch(Exception e)
        {
                Screen.showMessage("Error : ImportData - assignData : "+e);
        }
```

```
 }
}
```

Fuzzification:

```
import java.sql.*;
import java.util.*;
public class FuzzyConversion
{
        int slno = 0,ct = 0,ucsize = 0,ucshape = 0,ma = 0;
         int sece = 0,bn = 0,bc = 0,nn = 0,m = 0,class1 = 0,row = 0;
         int fct = 0,fucsize = 0,fucshape = 0,fma = 0;
         int fsece = 0,fbn = 0,fbc = 0,fnn = 0,fm = 0,fclass1 = 0;
        double cz = 0.0, co = 0.0, cb = 0.0;
        double fzvalues[];

          String pid = "";

        FuzzyConversion()
        {
                fzvalues = new double[30];
        }
         public void process()
        {
                try
                {
                        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
                        Connection con =
                           DriverManager.getConnection("jdbc:odbc:rulehide");
                        Statement stat = con.createStatement();
                        String sql = "Select * from diagnosis order by slno";
                        ResultSet rset = stat.executeQuery(sql);
                while(rset.next())
                {
                         slno = rset.getInt(1);
                         pid = rset.getString(2);
                         ct = rset.getInt(3);
                        cz = 0.0;
                         co = 0.0;
                         cb = 0.0;
                         getFuzzyValues((double)ct);
                         fzvalues[0] = cz;
                        fzvalues[1] = co;
                         fzvalues[2] = cb;
                         ucsize = rset.getInt(4);
                         cz = 0.0;
                          co = 0.0;
```

```
            cb = 0.0;
            getFuzzyValues((double)ucsize);
            fzvalues[3] = cz;
            fzvalues[4] = co;
            fzvalues[5] = cb;
            ucshape = rset.getInt(5);
            cz = 0.0;
            co = 0.0;
            cb = 0.0;
            getFuzzyValues((double)ucshape);
            fzvalues[6] = cz;
            fzvalues[7] = co;
            fzvalues[8] = cb;
            ma = rset.getInt(6);
            cz = 0.0;
            co = 0.0;
            cb = 0.0;
            getFuzzyValues((double)m);
            fzvalues[24] = cz;
            fzvalues[25] = co;
            fzvalues[26] = cb;
            class1 = rset.getInt(12);
            cz = 0.0;
            co = 0.0;
            cb = 0.0;
            getFuzzyValues((double)class1);
            fzvalues[27] = cz;
            fzvalues[28] = co;
            fzvalues[29] = cb;
            updateData();

            row++;
        }
    }catch(Exception e)
    {
        Screen.showMessage("Error : FuzzyConversion - process : "+e);
    }
}
public void updateData()
{
    try
    {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        Connection con = DriverManager.getConnection("jdbc:odbc:rulehide");
        String sql ="Insert into fuzzysets
            values(?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?)";
        PreparedStatement pstat = con.prepareStatement(sql);
        pstat.setInt(1,slno);
        pstat.setString(2,pid);
        for(int i=0;i<30;i++)
        pstat.setDouble((i+3),fzvalues[i]);
         pstat.executeUpdate();
         pstat.close();
          con.close();
  }catch(Exception e)
  {
        Screen.showMessage("Error : FuzzyConversion : updateData:"+e);
  }
}
public void getFuzzyValues(double idata)
{
        double a=0.0, b=2.0,c=4.0;
        if(idata<=a||c<=idata)
        {
                 cz = 0.0;
        }
        else
        {
                if(idata>=a && idata<=b)
                cz = ((idata-a)/(b-a));
                else if(idata>=b && idata<=c)
                cz = ((c-idata)/(c-b));
        }
        double a1=2.0, b1=5.0,c1=8.0;
        if(idata<=a1||c1<=idata)
        {
                co = 0.0;
        }
        else
        {
                if(idata>=a1 && idata<=b1)
                 co = ((idata-a1)/(b1-a1));
                else if(idata>=b1 && idata<=c1)
                co = ((c1-idata)/(c1-b1));
        }
        double a2=6.0, b2=8.0,c2=10.0;
        if(idata<=a2||c2<=idata)
        {
                cb = 0.0;
        }
        else
        {
                if(idata>=a2 && idata<=b2)
                cb = ((idata-a2)/(b2-a2));
        else if(idata>=b2 && idata<=c2)
                cb = ((c2-idata)/(c2-b2));
        }
}
}
```

Interest Rule:

```
import java.io.*;
import java.sql.*;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.table.*;

class FInterestRule extends JDialog implements ActionListener
{
 JLabel lbltitle,lblminsup,lblmincon,lblminint,lblrecord;
 JTextField txtminsup,txtmincon,txtminint,txtrecord;
 JScrollPane jsp;
 String avalue1 = "",avalue2 = "",type ="",str = "";
 int row = 0,acount = 0,acount1 =0,acount2 = 0;
 double support = 0,confidenc = 0,interest = 0,cpir = 0;
 String ssupport = "",sconfidenc = "",sinterest = "",scpir = "";
 String minsup = "", mincon = "", minint = "",sql = "";
 String rdata = "";
 JTable jtab;
 TableColumn tcol;
 Font afont,tfont,ofont;
 WindowListener wlist;
 JPanel panel;
 JButton btfind,btback;

 MainMenu mainmenu;

 FInterestRule(MainMenu mainmenu)
 {
  super(mainmenu,"Interesting Fuzzy Rule Identification process",true);
  this.mainmenu = mainmenu;
  panel = (JPanel)getContentPane();
  panel.setLayout(null);

  afont = new Font("MS Sans Serif",Font.BOLD,18);
  tfont = new Font("MS Sans Serif",Font.PLAIN,14);
```

```
ofont = new Font("MS Sans Serif",Font.BOLD,14);

int h = ScrollPaneConstants.HORIZONTAL_SCROLLBAR_AS_NEEDED;
int v = ScrollPaneConstants.VERTICAL_SCROLLBAR_ALWAYS;

lbltitle = new JLabel("<html><h2>Interesting Fuzzy Rule Identification
   process</html></h2>");
lbltitle.setSize(650,30);
lbltitle.setLocation(50,20);
panel.add(lbltitle);

lblminsup = new JLabel("Minimum Support");
lblminsup.setSize(150,30);
lblminsup.setLocation(90,100);
lblminsup.setFont(tfont);
panel.add(lblminsup);

txtminsup = new JTextField();
txtminsup.setSize(170,30);
txtminsup.setLocation(250,100);
txtminsup.setFont(tfont);
panel.add(txtminsup);

lblmincon = new JLabel("Minimum Confidence");
lblmincon.setSize(150,30);
lblmincon.setLocation(90,150);
lblmincon.setFont(tfont);

wlist = new WindowAdapter()
{
 public void windowClosing(WindowEvent we)
 {
  dispose();
 }
};
addWindowListener(wlist);
setSize(800,600);
setVisible(true);
}
public void actionPerformed(ActionEvent ae)
{
 if(ae.getSource() == btfind)
 {
  if(txtminsup.getText().trim().length() == 0)
   {
    Screen.showMessage("Enter the Minimum Support");
```

```
     txtminsup.requestFocus();
     return;
    }
    if(txtmincon.getText().trim().length() == 0)
    {
     Screen.showMessage("Enter the Minimum Confidence");
     txtmincon.requestFocus();
     return;
    }
    str = txtminsup.getText().trim();
    support = Double.parseDouble(str);
    str = txtmincon.getText().trim();
    confidenc = Double.parseDouble(str);
    clearTable();
    assignData();
  }
  if(ae.getSource() == btback)
  {
   dispose();
  }
 }
public void assignData()
{
 row = 0;
 try
 {
  Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
  Connection con = DriverManager.getConnection("Jdbc:Odbc:rulehide");
  String str1 = "select * from acview3 where supxy >=? and pconf1 >=? ";
  PreparedStatement pstat = con.prepareStatement(str1);
  pstat.setDouble(1,support);
  pstat.setDouble(2,confidenc);
  ResultSet rset = pstat.executeQuery();

  while(rset.next())
  {
   avalue1 = rset.getString(1);
   avalue2 = rset.getString(2);
   acount = rset.getInt(3);
   acount1 = rset.getInt(4);
   acount2 = rset.getInt(5);
   support = rset.getDouble(6);
   support = Tools.round(support,6);
   confidenc = rset.getDouble(9);
   confidenc = Tools.round(confidenc,6);
```

```
      jtab.setValueAt(""+(row+1),row,0);
      jtab.setValueAt(avalue1,row,1);
      jtab.setValueAt(avalue2,row,2);
      jtab.setValueAt(""+support,row,3);
      jtab.setValueAt(""+confidenc,row,4);
      row++;
     }
    txtrecord.setText(""+row);
   }catch(Exception e)
   {
     Screen.showMessage("Exception - FInterestRule - assignData : "+e);
   }
  }
  public void clearTable()
  {
   for(int i=0;i<row;i++)
     for(int j=0;j<5;j++)
      jtab.setValueAt("",i,j);
  }
}
```

Rule Hiding:

```
import java.sql.*;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.table.*;

class HideRule extends JDialog implements ActionListener
{
 JLabel lbltitle,lblrecord,lblconfidence;
 JTextField txtrecord,txtconfidence;
 JScrollPane jsp;
 String attr1 = "",attr2 = " ",type = " ";
 double support = 0,confidenc = 0,interest = 0,cpir = 0;
 int row = 0,rc = 0;
 JTable jtab;
 TableColumn tcol;
 Font afont,tfont,ofont;
 WindowListener wlist;
 JPanel panel;
 String str = "";
 String aname1 = "",aname2 = "";
 double tconfidence = 0,count = 0,confidence = 0,nsupport= 0;
 double nconfidence = 0,ncount = 0;
```

```
JButton  bthide, btintrule, btback;

MainMenu mainmenu;

HideRule(MainMenu mainmenu)
{
        super(mainmenu,"Rule Hiding Process",true);
        this.mainmenu = mainmenu;
         panel = (JPanel)getContentPane();
        panel.setLayout(null);

        afont = new Font("MS Sans Serif",Font.BOLD,18);
        tfont = new Font("MS Sans Serif",Font.PLAIN,14);
        ofont = new Font("MS Sans Serif",Font.BOLD,14);

        int h = ScrollPaneConstants.HORIZONTAL_SCROLLBAR_AS_NEEDED;
        int v = ScrollPaneConstants.VERTICAL_SCROLLBAR_ALWAYS;

        lbltitle = new JLabel("<html><h2>Rule Hiding Process</html></h2>");
        lbltitle.setSize(400,20);
        lbltitle.setLocation(300,30);
        panel.add(lbltitle);


        jtab = new JTable(10000,5);
        jsp = new JScrollPane(jtab,v,h);
         jtab.setSelectionMode(ListSelectionModel.SINGLE_SELECTION);
        jtab.setAutoResizeMode(JTable.AUTO_RESIZE_OFF);
         jtab.setFont(tfont);
         jtab.setRowHeight(20);

        tcol = jtab.getColumn("A");
        tcol.setHeaderValue("S.No");
        tcol.setPreferredWidth(50);
        tcol.setResizable(false);

         tcol = jtab.getColumn("B");
         tcol.setHeaderValue("Attribute1");
        tcol.setPreferredWidth(150);
        tcol.setResizable(false);

        tcol = jtab.getColumn("C");
        tcol.setHeaderValue("Attribute2");
        tcol.setPreferredWidth(150);
        tcol.setResizable(false);
```

```
 wlist = new WindowAdapter()
 {
      public void windowClosing(WindowEvent we)
      {
              dispose();
      }
 };
 addWindowListener(wlist);
 assignData();
 setSize(800,600);
 setVisible(true);
 }
 public void actionPerformed(ActionEvent ae)
 {
      if(ae.getSource() == bthide)
      {
              if(txtconfidence.getText().trim().length() == 0)
              {
                      Screen.showMessage("Enter the Minimum Confidence");
                      txtconfidence.requestFocus();
                      return;
              }
              str = txtconfidence.getText().trim();
              tconfidence = Double.parseDouble(str);
               hideProcess();
              Screen.showMessage(rc+" rules modified");
      }
      if(ae.getSource() == btintrule)
      {
              new HInterestRule(mainmenu);
      }
      if(ae.getSource() == btback)
      {
              dispose();
      }
 }
public void hideProcess()
{
      rc = 0;
      DBTools.clearTable("rulehide","fuzzyopt1");
      try
      {
              Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
              Connection con = DriverManager.getConnection("Jdbc:Odbc:rulehide");
              String sql = "Select * from fuzzyopt order by aname1,aname2";
              Statement stat = con.createStatement();
```

```
    ResultSet rset = stat.executeQuery(sql);
    while(rset.next())
    {
        aname1 = rset.getString(1);
        aname2 = rset.getString(2);
        support = rset.getDouble(3);
        count = rset.getDouble(4);
        confidence = rset.getDouble(5);
        if(confidence >= tconfidence)
        {
            nsupport = support + (0.05 * support);
            ncount = count + (0.3 * count);
            nconfidence = nsupport / ncount;
            System.out.println(confidence+" : "+nconfidence);
            rc++;
        }
        else
        {
            nsupport = support;
            ncount = count;
            nconfidence = confidence;
        }
        updateData();
     }
    }catch(Exception e)
    {
        Screen.showMessage("Error: HideRule - hideProcess:"+e);
    }
}
public void updateData()
{
    try
    {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        Connection con =
        DriverManager.getConnection("Jdbc:Odbc:rulehide","rulehide","rulehide"
    );
    String sql = "Insert into fuzzyopt1 values(?,?,?,?,?)";
    PreparedStatement pstat = con.prepareStatement(sql);
    pstat.setString(1,aname1);
    pstat.setString(2,aname2);
     pstat.setDouble(3,nsupport);
     pstat.setDouble(4,ncount);
    pstat.setDouble(5,nconfidence);
     pstat.executeUpdate();
    pstat.close();
```

```
        con.close();
       }catch(Exception e)
       {
               Screen.showMessage("Error: FuzzyOptimize: updateData:"+e);
       }
}
 public void assignData()
 {
       row = 0;
       try
       {
               Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
               Connection con = DriverManager.getConnection("jdbc:odbc:rulehide");
               String str = "select * from fuzzyopt1";
               Statement stat = con.createStatement();
               ResultSet rset = stat.executeQuery(str);
               while(rset.next())
               {
                       attr1 = rset.getString(1);
                       attr2 = rset.getString(2);
                       support = rset.getDouble(3);
                       confidenc = rset.getDouble(5);

               }
                       txtrecord.setText(""+row);
       }
       catch(Exception e)
       {
               Screen.showMessage("Exception - HideRule - assignData : "+e);
       }
 }
}
```

REFERENCES

1. Oliveira, Stanley R.M. and Zaïane, O.R., "Privacy preserving frequent itemset mining", In the Proc. of the 2nd IEEE International Conference on Privacy, Security and Data Mining, Australian Computer Society Inc. 43-54, 2002.

2. Yuhong Guo, "Reconstruction-Based Association Rule Hiding", SIGMOD 2007 Workshop on Innovative Database Research 2007(IDAR2007), Beijing, China, June 10, 2007

3. Aris Gkoulalas Divanis, Vassilios S. Verykios, "An Integer Programming Approach for Frequent Itemset Hiding", Proceedings of the 15th ACM international Conference on Information and Knowledge Management, ACM CIKM 2006, Arlington, Virginia, USA , Pages: 748 – 757, November 2006.

4. Natwichai, J., Li, X., and Orlowska, M.E., "A reconstruction based algorithm for classification rules hiding.", In Proc. Seventeenth Australasian Database Conference (ADC2006), Hobart, Tasmania, Australia, vol 49, pp. 49-58, 16-19 January 2006.

5. Ramakrishnan Srikant, Rakesh Agrawal, "Mining Quantitative Association Rules in Large Relational Tables", International Conference on Knowledge Discovery and data mining – proceedings of the 1996 ACM SIGMOD conference, Canada, pages 1-12, June 4- 6 ,1996

6. Stanley R. M. Oliveira and Osmar R.Za¨ıane, "Algorithms for Balancing Privacy and Knowledge Discovery in Association Rule Mining", Database Engineering and Applications Symposium, 2003 Proceedings. Seventh International Volume , Issue , 16-54 – 63, 18 July 2003.

8. Tolga BERBEROGLU and Mehmet KAYA, "Hiding Fuzzy Association Rules in Quantitative Data", Proceedings of the 3rd International Conference on Grid and Pervasive Computing - Workshops - Volume 00, Pages 387-392, 2008

9. Imielinski, T., Virmani, A., Abdulghani, A., "Datamine: Application programming interface and query language for database mining" , KDD-1996 Proceedings, 256-260, 1996.

10. Y. Saygın. V.S.Verkios, and C. Clifton, "Using unknowns to prevent Discovery of Associations Rules", SIGMOD Record 30(4): 45-54, December 2001.

11. T. P. Hong, C. S. Kuo, S. C. Chi, "Mining association rules from quantitative data", Intell. Data Anal. 3 (5) pp.363–376, 1999.

12. R. Agrawal, R. Srikant, "Fast algorithm for mining association rules", VLDB'94, Proceedings of 20th International Conference on Very Large Data Bases, Santiago de Chile, Chile, 487-499, September 12-15,1994.

13. http://www.anderson.ucla.edu/faculty/jason.frand/teacher/technologies/palace/data mining.htm

14. http://www.statsoft.com/textbook/stdatmin.html

15. http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/sbaa/ report.fuzzysets.html

16. http://www.centerforpbbefr.rutgers.edu/Jan11-2008%20papers/4-2.doc

17.http://mlearn.ics.uci.edu/databases/breast-cancer-isconsin/breast-cancer-wisconsin.data

18. http://www.mathworks.com

19. Arun and K.Pujari, "Data mining Techniques", University Press, First Edition, 2001.