Analysis of a Population of Diabetic Patients Databases in Weka Tool

P.Yasodha, M. Kannan

Abstract - Data mining is an important tool in many areas of research and industry. Companies and organizations are increasingly interested in applying data mining tools to increase the value added by their data collections systems. Nowhere is this potential more important than in the healthcare industry. As medical records systems become more standardized and commonplace, data quantity increases with much of it going unanalyzed. Taking into account the prevalence of diabetes among men and women the study is aimed at finding out the characteristics that determine the presence of diabetes and to track the maximum number of men and women suffering from diabetes with 249 population using weka tool. In this paper the data classification is diabetic patients data set is developed by collecting data from hospital repository consists of 249 instances with 7 different attributes. The instances in the Dataset are pertaining to the two categories of blood tests, urine tests. WEKA tool is used to classify the data and the data is evaluated using 10-fold cross validation and the results are compared.

Keywords- Data Mining, Diabetics data, Classification algorithm, Association algorithm Weka tool

1. Introduction

HE main focus of this paper is the classification of different types of datasets that can be performed to determine if a person is diabetic. The solution for this problem will also include the cost of the different types of datasets. For this reason, the goal of this paper is classifier in order to correctly classify the datasets, so that a doctor can safely and costeffectively select the best datasets for the diagnosis of the disease. The major motivation for this work is that diabetes affects a large number of the world population and it's a hard disease to diagnose. A diagnosis is a continuous process in which a doctor gathers information from a patient and other sources, like family and friends, and from physical datasets of the patient. The process of making a diagnosis begins with the identification of the patient's symptoms. The symptoms will be the basis of the hypothesis from which the doctor will start analyzing the patient.

This is our main concern, to optimize the task of correctly selecting the set of medical tests that a patient must perform to have the best, the less expensive and time consuming diagnosis possible. A solution like this one, will not only assist doctors in making decisions, and make all this process more agile, it will also reduce health care costs and waiting times for the patients. This paper will focus on the analysis of data from a data set called Diabetes data set.

2. Related Work

The few medical data mining applications as compared to other domains. [4] reported their experience in trying to automatically acquire medical knowledge from clinical databases. They did some experiments on three medical databases and the rules induced are used to compare against a set of predefined clinical rules. Past research in dealing with this problem can be described with the following approaches: (a) Discover all rules first and then allow the user to query and retrieve those he/she is interested in. The representative approach is that of templates [3]. This approach lets the user to specify what rules he/she is interested as templates. The system then uses the templates to retrieve the rules that match the templates from the set of discovered rules. (b) Use constraints to constrain the mining process to generate only relevant rules. [12] proposes an algorithm that can take item constraints specified by the user in the association rule mining process so that only those rules that satisfy the user specified item constraints are generated. This also does not work well because doctors often do not have any specific rules to mine. (c) Find unexpected rules. This approach first asks the user to specify his/her existing knowledge about the domain. The system finds those unexpected rules [5, 6, 13].

3. Data Mining In The Expert System

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, cuts 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. In our thesis, some data mining methods on socialdemographic data of the users were applied. Correction and actuality of the data is very important for data mining for diabetes.

3.1 Data Mining by WEKA Engine

Weka is a collection of machine learning algorithms for data mining tasks. The algorithms can either be applied directly to a dataset or called from your own Java code. Weka contains tools for data preprocessing, classification, regression, clustering, association rules, and visualization. It is also wellsuited for developing new machine learning schemes. Weka is developed by the University of Waikato. In our paper we used the Weka as data mining engine, and made a bridge between the Diabetes Expert System interface and Weka. No user needs to install Weka in his workstation, but it do already enough to be installed on server machine.

3.1.1 Classification

Classification is a data mining (machine learning) technique used to predict group membership for data instances. For example, you may wish to use classification to predict whether the weather on a particular day will be "sunny", "rainy" or "cloudy". Popular classification techniques include decision trees and neural networks.

Bayes Network Classifier

Bayes Network learning uses various searching algorithms and quality measures. This is the base module for a Bayes Network Classifier, and also provides data structures (network structure, conditional probability distributions, etc.) and facilities common to Bayes Network learning algorithms like K2 and B

eprocess Classify Cluster Associate	Select attributes Visual	ze								
lassifier										
Choose BayesNet -D -Q weka.dass	ifiers.bayes.net.search.l	ocal.K2P	1 -S BAVES -E	weka.dassifiers.b	iayes.net.es	timate.SimpleEsti	mator A 0.5			
est options	Classifier output									
🕒 Use training set	Summary									
Supplied test set	Composite Cines	ifind Te		1.00		66 2661				
Cross-validation Folds 10	Incorrectly Class	saified	Instances	84		33.7349				
O Decembran celt 0' 55	Kappa statistic			0.52	34					
O receivage spin / / / //	Mean absolute e	rror		0.10	31					
More options	Root mean squar	ed error		0.22	43					
	Relative absolu	te error		62.95	52 %					
iom) URINE SUGAR (PP) +	Root relative a	quared e	ror	78.69	19 %					
	I Total Number of	Instanc	23	249						
Start Stop	Ignored Class U	inknown I	istances		4					
esult list (right-click for options)	Detailed Ac	curacy B	Class ==	8						
3:32:45 - rules.ZeroR										
3:33:24 - bayes.BayesNet		TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class		
		0.888	0.197	0.772	888.0	0.826	0.918	Nil		
		0.615	0.091	0.64	0.615	0.627	0.897	1.0		
		0.419	0.117	0.429	0.419	0.424	0.805	0.5		
		0	0	0	0	0	0.282	0.2		
		0 545	0.025	0 75	0 545	0 692	0.66/	1.5		
		0.343	0.008	0.75	0.2	0.032	0.693	2.0		
		0	0.0022	0.757	0	0	0.218	0.0		
		0	0	0	0	0	0.21	Trace		
	Weighted Avg.	0.663	0.127	0.632	0.663	0.645	0.877			
	Confusion M	atrix ==								
	abcde	fg	1 1 <	classified	83					
	95 4 8 0 0	0 0	0 a .	Nil						
	10 32 7 0 3	0 0	0 b	1.0						
		_			_		5			D
ate										
K									Log 🗸	r
🗿 🙆 📋 D					T			• 18 18	() 1:34 PM	
										- C

J48 Pruned Tree

J48 is a module for generating a pruned or unpruned C4.5 decision tree. When we applied J48 onto refreshed data, we got the results shown as below on Figure2.

Classifier		
Choose 348 -C 0.25 -M 2		
est options	Classifier output	
🖯 Use training set		
A Sumied test set	J48 pruned tree	
Cross-vaidaton Folds 10		
Percentage split % 66	URINE SUGAR(F) = H11: H11 (176.0/70.0)	
More options	URINE SUGAR(E) = 0.5: 1.0 (31.0/10.0)	
	URINE SUGAR(E) = 1.5: 2.0 (5.0/2.0) UNIVE SUGAR(E) = mile mil (20.0/2.0)	
Nom) DTVF S (CLD/DD)	 UNLIE SUGAR(F) = 111: 111 (32.0/13.0) UDINE SUGAR(F) = 1.0: 2.0 /5.0/1.0) 	
nung on and accounting if y	Come population = 100 100 100	
Start Stop	Number of Leaves : 5	
Lesult list (right-click for options)		
3:32:45 - rules. ZeroR	Size of the tree : 6	
3:33:24 - bayes.BayesNet		
3:36:51 - trees.8FTree	Time polymous build models () assessed	
13:37:32 - trees.348	Time baken to build model: 6 Seconds	
13:38:47 - trees.348	www.Stratified.cross-validation.www.	
13:39:05 - trees.348	=== Summary ===	
13:39:30 - TEES.398	1	
242421 - 6003310	Correctly Classified Instances 150 60.24	1.8
	Incorrectly Classified Instances 99 39.75	9 %
	Kappa statistic 0.3958	
	Mean absolute error 0.1259	
	Root mean squared error 0.2535	
	Relative absolute error 76.8413 %	
	Root relative squared error 88.9411 %	
	Total Number of Instances 249	
	Ignored Class Unknown Instances 4	
	Detailed Accuracy By Class	
atus		
к		100 A

Figure2 J48 Tree

REP Tree

Fast decision tree learner. Builds a decision/regression tree using information gain/variance and prunes it using reduced-error pruning (with back fitting). Only sorts values for numeric attributes once. Missing values are dealt with by splitting the corresponding instances into pieces (i.e. as in C4.5).

Choose RtPTree #12-V0.000	-113-51-0-1								
st options	Classifier output								
Use training set	REFIree								Q
9 Supplied test set Set]								
Cross-validation Folds 10	: Nil (166/95) [83/47]								
Percentage split % 66	Size of the tree : 1								
More options	Time taken to build mod	el: 0.02 s	econds						
om) URINE SUGAR(PP)	=== Stratified cross-va	lidation =	-						-
	Summary								
Start	Correctly Classified In		107		42 0210				
sult list (right-dick for options)	_ Incorrectly Classified	Instances	142		57.0281				
01:28 - trees.REPTree	Kappa statistic		0		0110202				
	Mean absolute error		0.16	24					- 11
	Root mean squared error		0.28	5					- 11
	Relative absolute error		99.17	25 %					
	Root relative squared e	rror	99.98	51 %					
	Total Number of Instanc	es	249						
	Ignored Class Unknown I	nstances		4					
	Detailed Accuracy B	y Class ==							
	TD Date	FD Date	Dresision	Penall	E-Managere	BOC Area	(1		
	1 1	1	0.43	1	0.601	0 48	1711		
	0	0	0	0	0	0.48	1.0		
	0	0	0	0	0	0.467	0.5		
	0	0	0	0	0	0.05	0.2		
	0	0	0	0	0	0.412	1.5		
	0	0	0	0	0	0.458	2.0		
	0	0	0	0	0	0.498	nil		
	0	0	0	0	0	0.05	0.0		
US								_	
								Log	38°. ×

Figure3 REP tree

Random Tree

Class for constructing a tree that considers K randomly chosen attributes at each node. Performs no pruning. Also has an option to allow estimation of class probabilities based on a hold-out set (backfitting).

rocess Classify Cluster Associa	te Select attributes Vis	ualize								
sifer	Sector Contractor									
Choose RandomTree -K 0 -M	1.0-51									
t options	Classifier output									
) Use training set	Stratifie	d cross-va	lidation =							Q
Supplied test set Set	Sunnary -									
Cross-validation Folds 10	Compartin (1)	naified To		107		12 0710				
Darrantana milt N. 65	Incorrectly (1	lappifiad	Instances	142		52 0281	•			
recensycspic xo	Varma atatiat	ie .	THEORIDES	112	20	37.0201				
More options	Nean absolute	AFFOR		0.03	41					
	Bash man an	error		0.13	11					
-) I DEVE CLICAD/DD)	Robe mean aqu	area error		0.23	12 6					
inj orane ocean(in)	· Relative abst	Tube error		104.20	4/ 5 01 0					
Chart Dino	NOOT TELATIVE	squared e	rror	104.20	8 15					
2001 2009	Intal Sumper	or instanc	es	249	10					
ult list (right-click for options)	Ignored Class	Unknown 1	nscances		4					
01:28 - trees.REPTree	Datailad	lower P								
15:34 - trees.NBTree	=== Derailed	Recuracy B	y class ==	-						
22:59 - trees.BFTree		-	-	Provided and	Berr 11	P. Maxana	500 June	01		
25:53 - trees.RandomTree		IF Rate	IP Nate	Precision	Recall	I-Reasure	NUL Area	01655		
		0.333	0.901	0.439	0.935	0.03/	0.336	NII		
		0.035	0.025	0.375	0.055	0.1	0.344	1.0		
		0.07	0.019	0.429	0.07	0.12	0.507	0.5		
		0	0	9	0	0	0.159	0.2		
		0	0	0	0	0	0.45	1.5		
		0	0.004	0	0	0	0.527	2.0		
		0.05	0.009	0.333	0.05	0.087	0.493	nil		
		0	800.0	0	0	0	0.149	0.0		
		0	0	0	0	0	0.161	Trace		
	Weighted Avg.	0.43	0.397	0.368	0.43	0.305	0.529			
	Confusion	Matrix ==	-							
	a b c	d e	fah	i (c]	aggified	45				r
	100 2 1	0 0	1 2 1	0 a =	Wil	121				L
	47 3 2	0 0	0 0 0	0 b =	1.0					E.
us									-100-	
										Ø.
		۵.	-			-	-	· .	0 8:26 P	M

3.1.2 Association Rules

One of the reasons behind maintaining any database is to enable the user to find interesting patterns and trends in the data. For example, in a supermarket, the user can figure out which items are being sold most frequently. But this is not the only type of `trend' which one can possibly think of. The goal of database mining is to automate this process of finding interesting patterns and trends. Once this information is available, we can perhaps get rid of the original database. The output of the data-mining process should be a "summary" of the database. This goal is difficult to achieve due to the vagueness associated with the term `interesting'. The solution is to define various types of trends and to look for only those trends in the database.

Apriori

Apriori is a module implementing an Apriori-type algorithm. Iteratively reduces the minimum support until it finds the required number of rules with the given minimum confidence. The algorithm has an option to mine class association rules. It is adapted as explained in the second reference. Apriori is a classic algorithm for learning association rules. Apriori is designed to operate on databases containing transactions (for example, collections of items bought by customers, or details of a website frequentation). Other algorithms are designed for finding association rules in data having no transactions. As is common in association rule mining, given a set of itemsets (for instance, sets of retail transactions, each listing individual items purchased), the algorithm attempts to find subsets which are common to at least a minimum number C (the cutoff, or confidence threshold) of the itemsets. Apriori uses a "bottom up" approach, where frequent subsets are extended one item at a time (a step known as candidate generation), and groups of candidates are tested against the data. The algorithm terminates when no further successful extensions are found (Agrawal et al. 1994).

Action whole Account			
<pre>Accounts reduct To To</pre>	Apriori -N 10 -T 0 -C 0.9	0.05-01.0 40.1-5-1.0 ≪-1	
<pre>Active: 6 But But But But But But But But But But</pre>	Stro Associator out	ut	
<pre>press p</pre>	Attributes	: 6	
<pre>GT GENER ELCO STUDA(F) ELCO STUDA(F) URIE STUDA(F) ELLE STUDA(F) ELLE STUDA(F) === Associator model (full training set) === Apriori minimum metric confidence: 0.3 Hanker et guides performent: 13 Generated sets of large itematers: Size of set of large itematers 1(2): 0 Size of set of large itematers 1(2): 0 Size of set of large itematers 1(2): 0 Size of set of large itematers 1(2): 2 Bets rules found: 1. GENER-Market GUIDE STUDA(F)=GUIDE STUDA(F)=GUID St conf:(1) 2. GENER-Menule GUIDE STUDA(F)=GUIDE STUDA(F)=GUID St conf:(1.2) 3. GENER-Menule GUIDE STUDA(F)=GUID St conf:(1.2) 3. GENE</pre>	ht-click for a	NAME	
<pre>BLOC STUAR(F) BLOC STUAR(F) URLE STUAR(F) URLE STUAR(F) URLE STUAR(F) == & Secoistor model (full training set) == Apriori ===== Minimum settor confidence: 0.3 Bunker of option performat: 18 Generated sets of large itemets 1:8 Generated sets of large itemets 1:8 Generated sets of large itemets 1:9 Size of set of large itemets 1:1: 5 Size of set</pre>	niari	GENDER	
<pre>BLOC SUBA(P) URIE SUBA(P) URIE SUBA(P) URIE SUBA(P) === Associator model (full training set) === Apriori ===== Winimum seprort: 0.1 (/5 instances) Minimum metric confidences: 0.3 Banker of cycles performent: 18 Generated sets of large instances: Size of set of large instances: S</pre>		BLOOD STIGLE (F)	
<pre>units SUBA(P) units SUBA(P) units SUBA(P) units SUBA(P) units SUBA(P) units SUBA(P) units units u</pre>		RICON STICLE (DD)	
<pre>uBlE SDBA(P) uBlE SDBA(P) == Associator model (full training set) == Apriori memee Minimum septor: 0.1 (55 instances) Minimum settic confidences: 0.3 Runber of cycles performed: 18 Generated sets of large itemetes: Size of set of large item</pre>		IDINE SIGNATION	
<pre>windowskip == Associator model [full training set] == Apriori ==== Minimum seture confidences: 0.3 Dunker of cycles performes: 18 Generated sets of large itemates: Size of set of large itemates: Size of set of large itemates: Size of set of large itemates: 10: 2 Berr rules found: 1. cENDEMable (NILE SUBA(F)=dil 5: => (NILE SUBA(F)=dil 5: S conf:(1) 2. CENDEMable (NILE SUBA(F)=dil 5: =>> (NILE SUBA(F)=dil 5: S conf:(1) 2. CENDEMable (NILE SUBA(F)=dil 5: =>>> (NILE SUBA(F)=dil 5: S conf:(1) 3. CENDEMable (NILE SUBA(F)=dil 5: =>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>></pre>			
<pre>Applied Hole (Hile Statements) == Applied Hinima spect: 0.1 [15 instances) Hinima metric confidences: 0.3 Darker of cycles performed: 18 Generated sets of large itemets 1(1): 9 Size of set of large itemets 1(1): 9 Size of set of large itemets 1(1): 9 Size of set of large itemets 1(2): 8 Size of set of large itemets 1(2): 8 Size of set of large itemets 1(2): 2 Best rules found: 1. ESUB2-Metale HEINE SUBA(PF)=HIL 52 => HEINE SUBA(P)=HIL 52 conf:(1) 2. EHEE SUBA(P)=HIL 51 =>> HEINE SUBA(P)=HIL 54 conf:(1.59) 3. EHEEM-female HEINE SUBA(P)=HIL 55 =>> HEINE SUBA(P)=HIL 54 conf:(1.59)</pre>	and lance	unine suden (full twoining est)	
Apriori Minimu support 0.1 (25 instance) Minimu sector confidence: 0.3 Humber of cycles performes: 13 Generated sets of large itemates: Size of set of large itemates: Size of set of large itemates 1(2): 8 Size of set of large itemates 1(2): 8 Size of set of large itemates 1(2): 8 Size of set of large itemates 1(2): 2 Bers rules found: 1. GENER-Male (MINE STORA(F)=MIL 15 ==> VENDE STORA(F)=MIL 15 = conf:(1) 2. GENER-Menule WINE STORA(F)=MIL 55 ==> VENDE STORA(F)=MIL 154 = conf:(1.8) 3. GENER-Menule WINE STORA(F)=MIL 55 ==> VENDE STORA(F)=MIL 154 = conf:(0.8)	=== A330C.	ator model (full training set) ===	
<pre>Apriori Winimum support: 0.1 (35 instances) Size of set of large itemates 1(1): 9 Size of set of large itemates 1(1): 2 Best rules found: 1. GENER-Male Winis SUBA(9)-Will S2 =>/ WINIE SUBA(9)-Will S2 confr((1) 2. GENER-Male Winis SUBA(9)-Will S2 =>/ WINIE SUBA(9)-Will S4 confr((1) 3. GENER-Menale Winie SUBA(9)-Will S5 =>>/ WINIE SUBA(9)-Will S4 confr((1.8)) </pre>			
<pre>Himinm support: 0.1 (35 instances) Himinm serie confidences) 0.3 Barber of cycles performent: 18 Generated sets of large instances 1 Generated sets of large instances 1(1): 9 Size of set of large instances 1(2): 8 Size of set of large instances 1(2): 8 Size of set of large instances 1(2): 2 Best nules found: 1. GENER-Model GENE STORA(F)=dil 52 =>> URDE STORA(F)=dil 52 conf:(1) 2. GENER-Model GENE STORA(F)=dil 52 =>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>></pre>	Sectors!		
<pre>Minimum support: 0.1 (55 instances) Minimum matrix confidence: 0.3 Humber of option performat: 18 Generated sets of large itemates: Size of set of large itemates: Size of set of large itemates: Size of set of large itemates 1(2): 8 Size of set of large itemates 1(2): 2 Best rules found: 1. CENER-Male UNIX STUDA(F)=UNIX STUDA(F)=UNI</pre>	ADLIDLI		
<pre>Hinimus sepport: 0.1 (15 instance) Hinimus metric confidences: 0.3 Bunker of cycles performed: 18 Generated sets of large itemsets : Size of set of large itemsets 1(1): 9 Size of set of large itemsets 1(1): 9 Size of set of large itemsets 1(1): 2 Best rules found: 1. SERUES-MetalE NUME SYNAR(PF)=Mill 52 =>> URINE SYNAR(P=Mill 52 conf:(1) 2. GENES-MetalE NUME SYNAR(PF)=Mill 52 =>>> URINE SYNAR(P=Mill 54 conf:(1) 3. GENES-MetalE NUME SYNAR(PF)=Mill 55 =>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>></pre>			
<pre>Hinimum Perior confidence() i.3 Hinimum Perior confidence() i.3 Hinimum Perior confidence() i.3 Hinimum Perior Perio</pre>	Minimum au	mover. () 1 (25 instances)	
<pre>Hankes of cycles performed: 13 Hankes of groups performed: 13 Generated sets of large itemates: 13 Generated sets of large itemates 1(1): 9 Size of set of large itemates 1(1): 8 Size of set of large itemates 1(1): 8 Size of set of large itemates 1(1): 2 Best rules found: 1. GENUES-Main UNINE SUBAR(PS-Mill 52 ⇒> UNINE SUBAR(P-Mill 52 conf:(1) 2. GRUES-Menale UNINE SUBAR(PS-Mill 52 ⇒>> UNINE SUBAR(P-Mill 54 conf:(1.89) 3. GENUES-Menale UNINE SUBAR(PS-Mill 55 ⇒>>>>>>>>> UNINE SUBAR(P-Mill 54 conf:(1.89)</pre>	Minimum av	pport, tri (20 matalate)	
<pre>putter to typice private: if Generated sets of large itematic: Size of set of large itematics L(1): 9 Size of set of large itematics L(2): 8 Size of set of large itematics L(2): 8 Size of set of large itematics L(2): 2 Best nules found: 1. GENER-Media (RIME STORA(F)=dil 15 => URDE STORA(F)=011 15 conf:(1) 2. GENER-Media (RIME STORA(FP)=dil 15 => URDE STORA(F)=011 15 conf:(1) 3. GENER-Media (RIME STORA(FP)=011 15 =>> URDE STORA(F)=011 15 conf:(1.39) 3. GENER-Media (RIME STORA(FP)=011 15 =>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>></pre>	Number of	unic Commutences, 0.5	
<pre>Described sets of large itemates: Size of set of large itemates 1(1): 9 Size of set of large itemates 1(2): 8 Size of set of large itemates 1(2): 2 Dest rules found: 1. GENER-Male UNIE SUBA(P)=H11 S2 =>> UNIE SUBA(P)=H11 S2 confr((1) 2. UNIE SUBA(P)=H11 S2 =>>> UNIE SUBA(P)=H11 S4 confr((1) S3) 3. GENER-Female UNIE SUBA(P)=H11 S5 =>>> UNIE SUBA(P)=H11 S4 confr((1) S4) 3. GENER-Female UNIE SUBA(P)=H11 S5 =>>> UNIE SUBA(P)=H11 S4 confr((1) S4)</pre>	Dumber OF	ojoies periormen. 10	
<pre>endedted rea to tappe inductor Size of set of large inductor Size of set of large inductor L(): 8 Size of set of large inductor L(): 8 Size of set of large inductor L(): 2 Best rules found: 1. ESUBENHELE UNIT =>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>></pre>	Canaratad	eate of lawse itemaate:	
<pre>Size of set of large itemaets 1(1): 9 Size of set of large itemaets 1(1): 9 Size of set of large itemaets 1(3): 2 Bert rules found: 1. GENEE-Male UNITE SUBA(F)=011 52 =>> URIDE SUBA(F)=011 52 conf:(1) 2. GENEE SUBA(F)=011 107 =>> URIDE SUBA(F)=011 107 =>> URIDE SUBA(F)=011 54 conf:(0.39) 3. GENEE-Fenale URIDE SUBA(F)=011 55 =>> URIDE SUBA(F)=011 54 conf:(0.39)</pre>	Generated	bets of fatge formets.	
<pre>Size of set of large itemates 2(); 5 Size of set of large itemates 1(); 6 Size of set of large itemates 1(); 2 Best rules found: 1. CHIES-Media URIE STUDA((2)=Mil 52 =>> URIE STUDA((2)=Mil 52 conf:(1) 2. URIE STUDA((2)=Mil 107 =>> URIE STUDA((2)=Mil 106 conf:(1.59) 3. CHIES-Female URIE STUDA((2)=Mil 55 =>> URIE STUDA((2)=Mil 54 conf:(1.59) </pre>	Size of p	t of laves itemasts [/1]; 0	
<pre>Size of set of large itemmets 1(2): 8 Size of set of large itemmets 1(3): 2 Best nules found: 1. eSUES-Media (BINE 50004(F)=Hil 52 =>> URDE 50048(F)=Hil 52 conf:(1) 2. ENEE SOUGA(F)=Hil 107 =>> URDE 50048(F)=Hil 146 conf:(1.59) 3. ESUES-female GRDE 50048(F)=Hil 55 =>> URDE 50048(F)=Hil 54 conf:(1.59)</pre>	Size of St	o or range rounded b(r), s	
Size of set of large itemets 1(3): 2 Best rules found: 1. GENER-Male (KINE SUBA(?2)=H11 52 => URDE SUBA(?)=H11 52 conf:(1) 2. URDE SUBA(?2)=H51 107 => URDE SUBA(?)=H11 104 conf:(0.59) 3. GENER-Memale URDE SUBA(?2)=H11 55 => URDE SUBA(?)=H11 54 conf:(0.59)	Size of p	t of layns itempate 1/2). 8	
<pre>Size of set of large itemaets 1(3): 2 Best rules found: 1. SEDER=Maie URINE STORAR(PF)=UI1 52 =>> URINE STORAR(P)=UI1 52 conf:(1) 2. RUEE STORAR(PF)=UI1 107 =>> URINE STORAR(P)=UI1 54 conf:(1.59) 3. GENERA=female URINE STORAR(PP)=UI1 55 =>> URINE STORAR(P)=UI1 54 conf:(1.59)</pre>	JILE OL 5	o or rarge respects b(z). o	
<pre>site to struct a function (s); 1 Best rules found: 1. GENER-Male URIE SUBA(\$7)=011 52 =>> URIDE SUBA(\$7=011 52 conf:(1) 2. URIE SUBA(\$7\$=011 127 =>> URIDE SUBA(\$7\$=011 147 =>>> URIDE SUBA(\$7\$=011 147 =>>> URIDE SUBA(\$7\$=011 147 =>>> URIDE SUBA(\$7\$=011 147 =>>>>>>>>> URIDE SUBA(\$7\$=011 147 =>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>></pre>	Size of a	t of Javan itemate I(2): 2	
<pre>Best rules found: 1. CHUES-Mair UNIE STUDA(??)=011 52 => URINE STUDA(?)=011 52 confr(1) 2. URINE STUDA(??)=011 107 =>> URINE STUDA(?)=011 104 confr(0.39) 3. CHUES-Female URINE STUDA(??)=011 55 =>> URINE STUDA(?)=011 54 confr(0.59)</pre>	DITE OF D	o or range romaena s(s), z	
1. SUBER-HENG HENE SUBA(FF)-HI1 52 ⇒> HENE SUBA(F)-HI1 52 conf:(1) 2. HENE FEB348 (FF)-HI1 107 ⇒> HENE SUBA(F)-HI1 104 conf:(0.39) 3. SENER-Female HENE SUBA(FF)-HI1 55 ⇒> HENE SUBA(F)-HI1 54 conf:(0.39)	Rest viller	founds	
 CHUES-Male (RINE STORA(PP+Hi1 52 ⇒> URINE STORA(P+Hi1 52 conf:(1) URINE STORA(PP+Hi1 107 ⇒> URINE STORA(P+Hi1 104 conf:(0.8) SEUER-Female (RINE STORA(PP+Hi1 55 ⇒> URINE STORA(P+Hi1 54 conf:(0.8)) 	0030 1010	Louis.	
2. HRIE SUBAR(PF=HD1 107 →> HRIE SUBAR(PF=HD1 104 confr(0.59) 3. GENERA-Female HRIE SUBAR(PF=HD1 55 →> UBIE SUBAR(Pf=HD1 54 confr(0.58)	1 GENDES	=Male HRINF SHCLR/DP(=Nil 52 ==> HRINF SHCLR/F(=Nil 52 conf:(1)	
<pre>s. estudie conductive (set in the set of the set</pre>	2 101102	STICLD / DOI - Will 1 / /> TID THE STICLD / F) - Will 1 / 6 / confr (/ 0 00)	
	3 CENTRE	-Female HDTHF SHCAD (DO)-HS1 55> HDTHF SHCAD (F)-HS1 54 conf-(A 0P)	
	o. centres	-conte onthe contract (and an> onthe contract of contractory	
Log			Log

Figure5 Apriori

4. **RESULTS & DISCUSSIONS**

The main purpose of the system is to guide diabetic patients during the disease. Diabetic patients could benefit from the diabetes expert system by entering their daily glucoses rate and insulin dosages; producing a graph from insulin history; consulting their insulin dosage for next day. The diabetes expert system is not only for diabetic patient, but also for the people who suspect if they are diabetic. It's also tried to determine an estimation method to predict glucose rate in blood which indicates diabetes risk. In our Paper, the target class consists of 249 men and women who are actually affected with diabetes. However, it is unknown that in what stage the disease is. Hence this study will not only help in estimating the maximum number of people suffering from diabetes with specific characteristics but also helps in identifying the state of the disease.

REFERENCES

[1] Mats Jontell, Oral medicine, Sahlgrenska Academy, Göteborg University (1998) "A Computerised Teaching Aid in Oral Medicine and Oral Pathology. "Olof Torgersson, department of Computing Science, Chalmers University of Technology, Göteborg.

[2] T. Mitchell, "Decision Tree Learning", in T. Mitchell, Machine Learning (1997) the McGraw-Hill Companies, Inc., pp. 52-78.

[3] Klemetinen, M., Mannila, H., Ronkainen, P., Toivonen, H., and Verkamo, A. I (1994) "Finding interesting rules from large sets of discovered association rules," CIKM.

[4] Tsumoto S., (1997)"Automated Discovery of Plausible Rules Based on Rough Sets and Rough Inclusion," Proceedings of the Third Pacific-Asia Conference (PAKDD), Beijing, China, pp 210-219.

[5] Liu B., Hsu W., (1996) "Post-analysis of learned rules," AAAI, pp. 828-834.

[6] Liu B., Hsu W., and Chen S., (1997) "Using general impressions to analyze discovered classification rules," Proceedings of the Third ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.

[7] Stutz J., P. Cheeseman. (1996) Bayesian classification (autoclass): Theory and results. In Advances in Knowledge Discovery and Data Mining. AAAI/MIT Press

[8] Witten Ian H., E. Frank, Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations, Ch. 8, © 2000 Morgan Kaufmann Publishers

[9] http://www.cs.waikato.ac.nz/ml/weka/, accessed 06/05/21.

[10] <u>http://grb.mnsu.edu/grbts/doc/manual/</u> J48_Decision_T rees.html, accessed

[11] Wikipedia, ID3-algorithm (accessed 2007/12/09) (URL: http://en.wikipedia.org/wiki/ID3_algorithm)

[12] Srikant,R.,Vu,Q.andAgrawal,R.,(1997), "Mining association rules with item constraints," Proceedings of the Third International Conference on Knowledge Discovery and Data Mining, Newport Beach, USA, pp 67-73.

AUTHOR PROFILE:



P.Yasodha Mphil Research Scholar in the Department of Computer Science & Applications in Sri Chandrasekharendra Saraswathi Viswa Mahavidyalaya University, Enathur, Kanchipuram –631 561. She received the degree in Master of Computer Applications from SCSVMV University in 2010. Her research interest lies in the area of Data Mining and Artificial Intelligence.



M.Kannan has been working as a Assistant Professor and Ph.D Research Scholar in the Department of Computer Science & Applications in Sri Chandrasekharendra Saraswathi Viswa Mahavidyalaya University, Enathur, Kanchipuram -631 561. He received the degree in Master of Computer Applications from Bharathidasan University in 2001 and M.Phil(Computer Science) from Madurai Kamaraj University in 2005. His research interest includes Software Engineering & Data Mining.