

Adapting Association Rules Mining to task of Classification

Prof.Prachitee Shekhawat

Alard College of Engineering and Management,Pune,Maharashtra,India.

Abstract

Classification rule mining is used to discover a small set of rules in the database to form an accurate classifier. Association rules mining are used to reveal all the interesting relationship in a potentially large database. These two techniques can be integrated to form a framework called Associative Classification method. The integration is done by focusing on mining a special subset of association rules called Class Association rules (CAR). This project paper proposes a Neural Network Association Classification system, which is one of the approaches for building accurate and efficient classifiers. Experimental result shows that the classifier build for diabetes dataset in this way is more accurate than previous Classification Based Association.

Keywords: Data Mining, Association Rule Mining, Classification, Associative Classification, Backpropagation neural network.

1. Introduction

Classification rule mining and Association rules mining are the two important data mining techniques. Classification rule mining is used to discover a small set of rules in the database to form an accurate classifier. Association rules mining are used to reveal all the interesting relationship in a potentially large database. Association rule mining finds all rules in the database that satisfy some minimum support and minimum confidence threshold. For association rule mining, the target of the discovery is not predetermined, while for classification rule mining there is one and only one predetermined target. These two techniques can be integrated to form a framework called Associative Classification method.

The integration is done in order to get a special subset of association rules whose right-hand re restricted to classification class attribute. These subsets of rules are referred as Class Association Rules.

The use of association rules for classification is restricted to problems where the instances can only belong to a discrete number of classes. The reason is that association rule mining is only possible for categorical attributes. The head Y of an arbitrary association rule $X \rightarrow Y$ is a disjunction of items. However, association rules in their general form cannot be used directly. We have to restrict

their definition. Every item which is not present in the rule body may occur in the head of the rule. When we want to use rules for classification, we are interested in rules that are capable of assigning a class membership. Therefore we restrict the head Y of a class association rule $X \rightarrow Y$ to one item. The attribute of this attribute-value-pair has to be the class attribute. A class association rule is obviously a predictive task. By using the discriminative power of the Class Association Rules we can also build a classifier.

Data mining in the proposed, Neural Network Associative Classification, system thus consists of three steps:

- 1) If any, discretizing the continuous attributes,
- 2) Generating all the Class Association Rules (CARs), and
- 3) Building a classifier with the help of Backpropagation Neural Network based on the generated CARs set.

Here, we have to analyze the diabetes dataset and mine all the accurate association rule that will be use to build an efficient classifier on the basis of predicted attributes. The motivations of choosing this dataset are:

- 1) China and India have the maximum number of diabetics' patient. It is mainly because of their huge population size and rapidly changing lifestyle over the last two decades. With around 90 million diabetics in China and 61.3 million diabetics patient in India.
- 2) Over time, high blood glucose damages nerves and blood vessels, leading to complications such as heart disease, stroke, kidney disease, blindness, dental disease, and amputations.
- 3) Adequate treatment of diabetes is thus important, as well as blood pressure control and lifestyle factors such as smoking cessation and maintaining a healthy body weight.

Diabetes dataset refers to 2 classes of 768 instances each. The 768 instances, which are separated between the 2 classes, contain the following predicted attributes: Number of times pregnant, Plasma glucose concentration, Diastolic blood pressure, Triceps skin fold thickness, serum insulin, Body mass index, Diabetes pedigree function and Age. Two classes are one contain patient that are tested positive for diabetes and another one who are negative. So, each instance will belong to one of the two classes: Positive or negative.

The paper is organized as follows: section 2 contains the brief introduction to major previous work done about data mining. Section 3 describes the proposed system, Neural Network Associative Classification, architecture and section 4 presents our experimental setup and discussed the result. Section 5 concludes the paper.

2. LITERATURE SURVEY

The data analysis algorithms (or data mining algorithms, as they are more popularly known nowadays) can be divided into three major categories based on the nature of their information extraction [1]:

- Clustering (also called segmentation or unsupervised learning),
- Predictive modeling (also called classification or supervised learning), and
- Frequent pattern extraction.

Clustering is the major class of data mining algorithms. The goal of the search process used by these algorithms is to identify all sets of similar examples in the data, in some optimal fashion. One of the oldest algorithms for clustering is k-means [2]. The two disadvantages of this algorithm are initialization problem and that the cluster must be linearly separable. To deal with the initialization problem, the global k-means has been proposed [3], which is an incremental-deterministic algorithm that employs k-means as a local search procedure. Kernel k-means algorithm [4] avoids the limitation of linearly separable clusters and it mapped the data points from input space to a higher dimensional feature through a nonlinear transformation ϕ and the k-means is applied in the feature space. Global kernel k-means [5] is an algorithm which mapped data points from input space to a higher dimensional feature space through the use of a kernel function and optimizes the clustering error in the feature space by locating near-optimal solution. Because of its deterministic nature, this makes it independent of the initialization problem, and the ability to identify nonlinearly separable cluster in input space. So global kernel k-means algorithm combines the advantages of both global k-means and kernel k-means. Another approach for clustering data is hierarchical clustering that is based on the Hungarian method [6] and the computational complexity of the proposed algorithm is $O(n^2)$.

The important classification algorithms are decision tree, Naive-Bayes classifier and statistics [2]. They use heuristic search and greedy search techniques to find the subsets of rules to find the classifiers. C4.5 and CART are the most well-known decision tree algorithms.

The final class of data mining algorithms is frequent pattern extraction. For a large databases, [7] describes an apriori algorithm that generate all significant association rules between items in the database. The algorithm makes the multiple passes over the database. The frontier set for a pass consists of those itemsets that are extended during the pass. In each pass, the support for candidate itemsets, which are derived from the tuples in the databases and the itemsets contain in frontier set, are measured. Initially the frontier set consists of only one element, which is an empty set. At the end of a pass, the support for a candidate itemset is compared with the minsupport. At the same time it is determined if an itemset should be added to the frontier set for the next pass. The algorithm terminates when the frontier set is empty. After finding all the itemsets that satisfy minsupport threshold, association rules are generated from that itemsets.

Bing Liu and et al.[8] had proposed an Classification Based on Associations (CBA) algorithm that discovers Class Association Rules (CARs). It consists of two parts, a rule generator, which is called CBA-RG, is based on Apriori algorithm for finding the association rules and a classifier builder, which is called CBA-CB. In Apriori Algorithm, itemset (a set of items) were used while in CBA-RG, ruleitem, which consists of a condset (a set of items) and a class. Class Association Rules that are used to create a classifier in [8] is more accurate than C4.5 [2] algorithm. But the Classification Based on Associations (CBA) algorithm needs the ranking rule before it can create a classifier. Ranking depends on the support and confidence of each rule. It makes the accuracy of CBA less precise than Classification based on Predictive Association Rules.

Neural network is a parallel processing network which generated with simulating the image intuitive thinking of human, on the basis of the research of biological neural network according to the features of biological neurons and neural network and by simplifying, summarizing and refining[9]. It uses the idea of non-linear mapping, the method of parallel processing and the structure of the neural network itself to express the associated knowledge of input and output. Initially, the application of the neural network in data mining was not optimistic, because neural networks may have complex structure, long training time, and uneasily understandable representation of results. But its advantages such as high affordability to the noise data and low error rate, the continuously advancing and optimization of various network training algorithms, especially the continuously advancing and improvement of various network pruning algorithms and rules extracting algorithm, make the application of the neural network in the data mining increasingly favored by the overwhelming majority of users. Xianjun Ni [10] describes Data mining process based on neural network. This process is composed of three main steps as data preparation, rule extraction and rules assessment.

3. NEURAL NETWORK ASSOCIATIVE CLASSIFICATION SYSTEM

The proposed system, Neural network Associative Classification, consists of three phases: Discretization Phase, Generating Class Association Rule Phase and the Classification phase. Before first phase, that is discretization phase, we have to transform the character data (if any) into numeric data. This is essential as neural network can only handle the numeric data so it is needed.

3.1. Discretization Phase

Classification datasets often contain many continuous attributes. Mining of association rules with continuous attributes is still a research issue. So our system involves discretizing the continuous attributes based on the pre-determined class target. For discretization, we have used the algorithm from [10].

3.2. Generating Class Association Rules Phase

This step presents a way for generating the Class Association Rules (CARs) from the pre-processed dataset.

3.2.1 Association Rules

Consider:

$D = \{d_1, d_2, \dots, d_n\}$ is a database that consist of set of n data and $d \subseteq I$

$I = \{i_1, i_2, \dots, i_m\}$ is a set of all items that appear in D

The association rule has a format, $A \rightarrow B$, by support=s%, confidence=c% and $A \cap B = \emptyset$ and where $A, B \subseteq I$.

- Support value is frequency of number of data that consist of A and B or $P(A \cup B)$ and is given by

$$\text{support}, s(A \rightarrow B) = \frac{\sigma(A \cup B)}{N} * 100\% \quad \dots(1)$$

- Confidence is frequency of number of data that consist of A and B or $P(B|A)$ and given by

$$\text{confidence}, c(A \rightarrow B) = \frac{\sigma(A \cup B)}{\sigma(A)} * 100\% \quad \dots(2)$$

3.2.2 Class Association Rules

The Class Association Rules are the subset of the association rules whose right-hand-side is restricted to the pre-determined target. According to this, a class association rule is of the form

$$A \rightarrow b_i$$

where b_i is the class attribute and A is $\{b_1, \dots, b_{i-1}, b_{i+1}, \dots, b_n\}$.

Consider:

$D = \{d_1, d_2, \dots, d_i\}$ is a database that consist of set of data that have n attribute and class label by $d = \{b_1, b_2, \dots, b_n, y_k\}$ where $k = 1, 2, \dots, m$.

$I = \{A_1, A_2, \dots, A_m\}$ is a set of all items that appear in D

$Y = \{b_1, b_2, \dots, b_m\}$ is a set of class label m class

A class association rule (CAR) is an implication of the form $\text{condset} \rightarrow b$, where $\text{condset} \subseteq I$, and $b \subseteq Y$. A rule $\text{condset} \rightarrow b$ holds in D with support = s% and confidence = c% following these conditions:

- Support value is the frequency of number of all data that consist of itemset condset and class label b.
- Confidence value is the frequency of number of data that consist of class label b when data have itemset condset .

Before formulating the formulas for support and confidence, few notations are defined as:

- condsupCount is the number of data that consist of condset .
- rulesupCount is the number of data that consist of condset and has the label b.

Then

$$\text{support} = \frac{\text{rulesupCount}}{|D|} * 100\% \quad \dots(3)$$

$$\text{confidence} = \frac{\text{rulesupCount}}{\text{condsupCount}} * 100\% \quad \dots(4)$$

The algorithm 1 is shown below which is used to find the Class Association Rules:

```

1:  $F_1 = \{\text{large 1-rule items}\}$ ;
2:  $CAR_1 = \text{genRules}(F_1)$ ;
3: for ( $k=2; F_{k-1}=\emptyset; k++$ )
4:    $C_k = \text{candidateGen}(F_{k-1})$ ;
5:   for each data case  $d \in D$ 
6:      $C_d = \text{ruleSubset}(C_k, d)$ ;
7:     for each candidate  $c \in C_d$ 
8:        $c.\text{condsupCount}++$ ;
9:       if  $d.\text{class}=c.\text{class}$  then
10:         $c.\text{rulesupCount}++$ ;
11:       end
12:     end
13:   end
14:    $F_k = \{c \in C_k \mid c.\text{rulesupCount} \geq \text{minsup}\}$ ;
15:    $CAR_k = \text{genRules}(F_k)$ ;
16: end
17:  $CARs = \bigcup_k CAR_k$ ;
    
```

Algorithm 1: Proposed Algorithm

An example of searching the Class Association Rules following the Algorithm 1 is given below in Table

Table 1: Example data.

	A	B	C
1.	a1	b1	c1
2.	a1	b2	c1
3.	a2	b1	c1
4.	a3	b2	c2
5.	a2	b2	c2
6.	a1	b1	c2
7.	a1	b1	c1
8.	a3	b3	c1
9.	a3	b3	c2
10.	a2	b3	c2

- A and B are predicted attributes. A have the possible values of a1, a2, and a3 and B's possible values are b1, b2, and b3. C is class label and possible values are c1 and c2.
- Minimum support(minsup) = 15%
- Minimum confidence(minconf) = 60%

In cases where a ruleitem is associated with multiple classes, only the class with the largest frequency is considered by current associative classification methods.

1 st pass	F ₁	<{(A,a1)},4),(C,c1),3>
		<{(A,a2)},3),(C,c2),2>
		<{(A,a3)},3),(C,c2),2>
		<{(B,b1)},4),(C,c1),3>
		<{(B,b2)},3),(C,c2),2>
		<{(B,b3)},3),(C,c2),2>
CAR ₁	(A,a1)→(C,c1)	
	(A,a2)→(C,c2)	
	(A,a3)→(C,c2)	
	(B,b1)→(C,c1)	
	(B,b2)→(C,c2)	
	(B,b3)→(C,c2)	
2 nd pass	C ₂	<{(A,a1),(B,b1)},(C,c1)>
		<{(A,a1),(B,b2)},(C,c1)>
		<{(A,a2),(B,b1)},(C,c1)>
		<{(A,a2),(B,b2)},(C,c2)>
		<{(A,a2),(B,b3)},(C,c2)>
		<{(A,a3),(B,b2)},(C,c2)>
		<{(A,a3),(B,b3)},(C,c1)>
		<{(A,a3),(B,b3)},(C,c2)>
	F ₂	<{(A,a1),(B,b1)},3),(C,c1),2>
CAR ₂	{(A,a1),(B,b1)} → (C,c1)	
CARs	CAR ₁ ∪ CAR ₂	

Table 2: Generating CARs following proposed algorithm

We can express details as in Table 2. For CAR, condset →y, the format for ruleitem is

$$\langle(\text{condset}, \text{condsupCount}), (y, \text{rulesupCount})\rangle$$

Consider for example the ruleitem $\langle\{(A,a1)\},(C,c1)\rangle$, it means that the frequency of condset (A,a1) in data is 4, it is also called as condsupCount, and the frequency of (A,a1) and (C,c1) occurring together is 3, also known as rulesupCount. So, the support and confidence are 30% and 75% respectively by using equations 3 and 4. As it satisfy the minsup threshold ruleitem is a frequent 1-ruleitem (F₁). As the confidence of this ruleitem is greater than minconf threshold, Class Association Rule (CAR₁) is formed as

$$(A, a1) \rightarrow (C, c1)$$

Repeat this procedure and generate all F₁ ruleitems. Using F₁ ruleitem form Candidate 2-ruleitems C₂ in the same way. Class Association Rules that are generated from table 1 is shown in table 2 and it can be express as:

- a. (A,a1) → (C, c1) s=30%, c=75%
- b. (A,a2) → (C, c2) s=20%, c=66.67%
- c. (A,a3) → (C, c2) s=20%, c=66.67%

- d. (B,b1) → (C, c1) s=30%, c=75%
- e. (B,b2) → (C, c2) s=20%, c=66.67%
- f. (B,b3) → (C, c2) s=20%, c=66.67%
- g. {(A,a1),(B,b1)} → (C, c1) s=20%, c=66.67%

where s and c are support and confidence value respectively.

1. Classification Phase

Backpropagation Neural Network is a feedforward neural network which is composed of a hierarchy of processing units, organized in a series of two or more mutually exclusive sets of neurons or layers. The first layer is an input layer which serves as the holding site for the inputs applied to the neural network. The basic role of this layer is to hold the input values and to distribute these values to the units in the next layer. The last, output, layer is the point at which overall mapping of the network input is available. Between these two layers lies zero or more layer of hidden units. In this internal layer, addition remapping or computing takes place.

Generally Backpropagation Neural Network use logistic sigmoid activation function when output is required in the range [0, 1]. The output at each node can be defined in this equation.

$$output_j = \sigma(net_j) = \frac{1}{(1+e^{-net_j})} \quad \dots (5)$$

By

$$net_j = \sum_{i=0}^n w_{ji} x_i \quad \dots (6)$$

Where, w_{ji} is the connection strength from the node i to node j and x_i is the output at node i .

The structure of Backpropagation neural network that is created with input as Class Association Rules is shown in fig. 2.

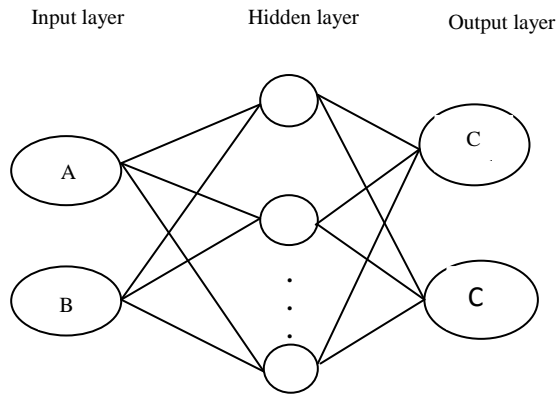


Fig. 2: Structure of Backpropagation Neural Network from Class Association Rules

Suppose that ruleitem for learning is $\{(A, a1), (B, b1)\} \rightarrow (C, c1)$. These character data need to be transformed into numeric data and so A with values $a1, a2,$ and $a3$ are encoded with numeric value as 1, 2, and 3 respectively. Similarly B with values $b1, b2,$ and $b3$ are encoded with numeric value as 1, 2, and 3 respectively. The class label $c1$ is denoted by "1 0" and $c2$ by "0 1" and now we will have the input as "1 1" and output will be "1 0" for the above mention ruleitem. This is shown in fig. 3.

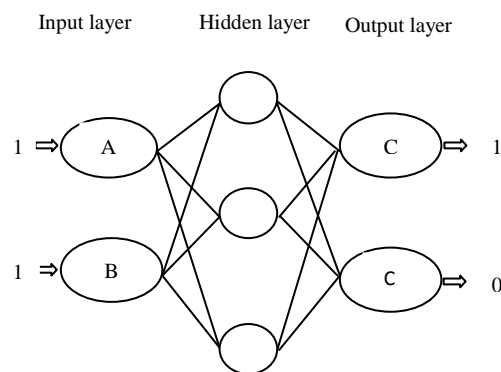


Fig. 3: Structure of Backpropagation Neural Network for $\{(A, a1), (B, b1)\} \rightarrow (C, c1)$ as an input.

4. Experimental setup and results

The Neural Network Associative Classification system is an application developed in order to classify the relational dataset. This system is essentially a process consisting of following operations

- 1) Discretizing continuous attributes, if any

- 2) System perform the operations of calculating CAR_s (Class Association Rules), based on two threshold values, support and confidence,

- 3) Let the system form a neural network, with the inputs as CAR_s set, by using Backpropagation algorithm, and perform testing on the network, and

- 4) User can enter the unknown input and the system will provide the predicted class based on the training.

The discriminating feature of this system is that it uses the CAR_s to train the network to perform the classification task. So the system will render the efficient and accurate class based on the predicted attributes.

4.1 Extracting Class Association Rules

First, finding the Class Association Rules (CAR) from diabetes datasets can be useful in many contexts. In general understanding the CAR_s relates the class attribute with the predicted attributes. Extraction of association rules was achieved by using the proposed algorithm in algorithm 1. The rules are extracted depending on the relations between the predicted and class attributes. The dataset in the fig. 4 is a segment of diabetes dataset, where class value 1 is interpreted as "tested positive for diabetes" and 2 is interpreted as "Tested Negative for diabetes"

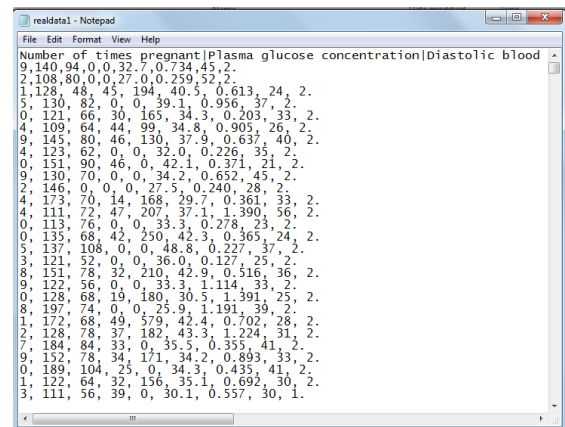


Fig. 4: An excerpt from an iris plant dataset

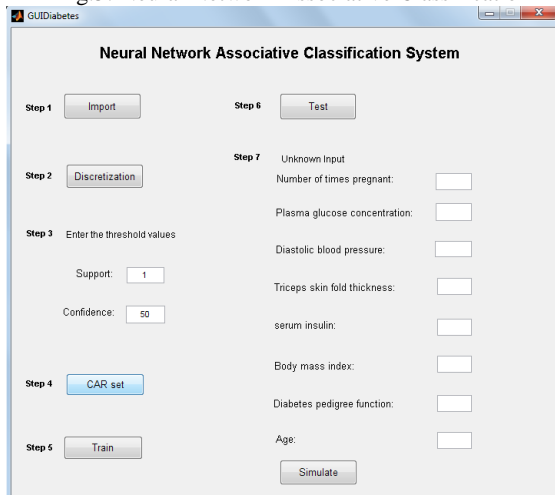
The aim is to use the proposed algorithm, to find the relations between the features and represents them in CAR_s to form the input to the backpropagation neural network. The two classes of Diabetes were allocated numeric representation as shown in table 4.

Table 4: Class and their numeric representation

Class	Numeric representation
Tested Positive	1 0
Tested Negative	0 1

Fig. 5 shows the snapshot of the main screen of our project. The import button will load the Diabetes dataset into the MATLAB workspace. As the attributes are continuous, discretization is done.

Fig.5: Neural Network Associative Classification



System

With the user define support 1% and confidence 50%, CARs are extracted by using Algorithm 1 and are shown in fig. 6, where s denotes support and c denotes confidence for that respective rule. Total 3011 rules are generated from the diabetes dataset.

4.2 Methodology for building the classifier

For training the Backpropagation neural network CARs are used. To train the network, we have used the Matlab function train(). In which 60% of data are used for training, 20% are used for testing and 20% for validation purpose. traingdm, learnGDM and tansig are used as a training function, learning function and transfer function. Number of hidden layer nodes =2000 and learning will stop when error is less than 0.005 or epochs= 230 times. These values are kept constant in order to determine momentum constant (mc) and learning rate (lr). The table 5 shows the number of misidentified testing patterns when momentum constant (mc) is varied and keeping learning rate constant to 0.1. The value for mc is selected for which misidentified testing patterns is minimum. The best performance is achieved when mc = 0.8.

Now, the values of learning rate are varied from 0.1 to 0.9 and mc is kept constant at 0.8 and numbers of misidentified testing patterns are shown in Table 6. So, best performance can be achieved at 0.1.

Table 5: Performance of Momentum Constant (mc)

Learning rate	Momentum constant	Misidentified testing patterns
0.1	0.1	153
	0.2	140
	0.3	211
	0.4	145
	0.5	171
	0.6	145
	0.7	174
	0.8	125
	0.9	144

Table 6: Performance of Learning Rate

Momentum constant	Learning Rate	Misidentified testing patterns
0.8	0.1	125
	0.2	160
	0.3	177
	0.4	163
	0.5	135
	0.6	159
	0.7	164
	0.8	175
	0.9	155

Now, to determine appropriate number of epochs, keep the values of all the other properties constant. Table 7 shows the best properties for Backpropagation neural network in order to build a classifier for Diabetes

Table 7: Properties to build Backpropagation neural network for Diabetes

Neural Properties network	Value
Number of hidden neurons	2000
Transfer function	Logsig
Learning rate	0.1
Momentum constant	0.8
Training technique	Scaled conjugate gradient Backpropagation
Epochs	230
Misidentified patterns	125
Training patterns	1807
Validation patterns	602
Testing patterns	602

Fig. 7 and 8 shows the snapshot of the training and testing respectively with the properties shown in table 7.

Now we have a trained network, which can more accurately predict the class based on the predicted attributes.

It can be observed that extracted CARs include the most important features and informative relations from the dataset which will help neural network form the accurate classifier for diabetes dataset.

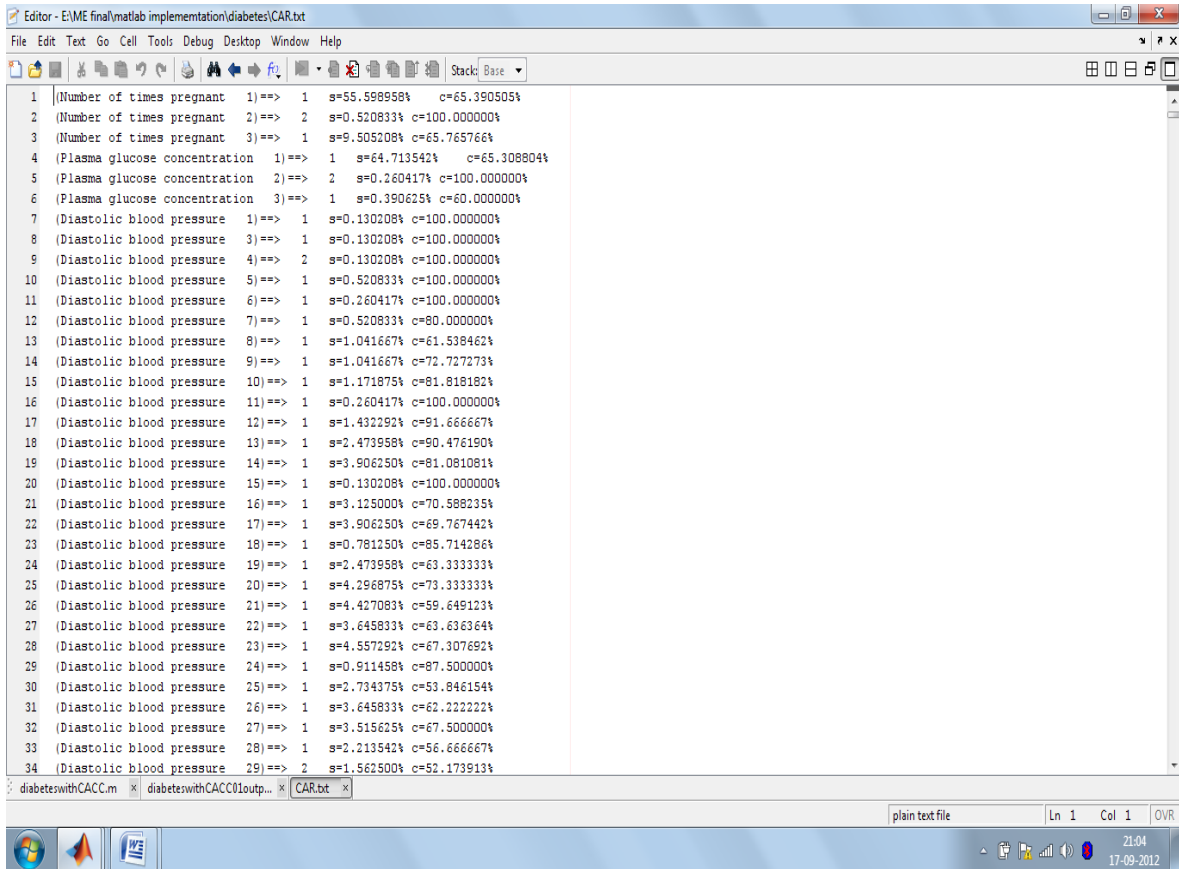


Fig. 6. Extracted Class Association Rules

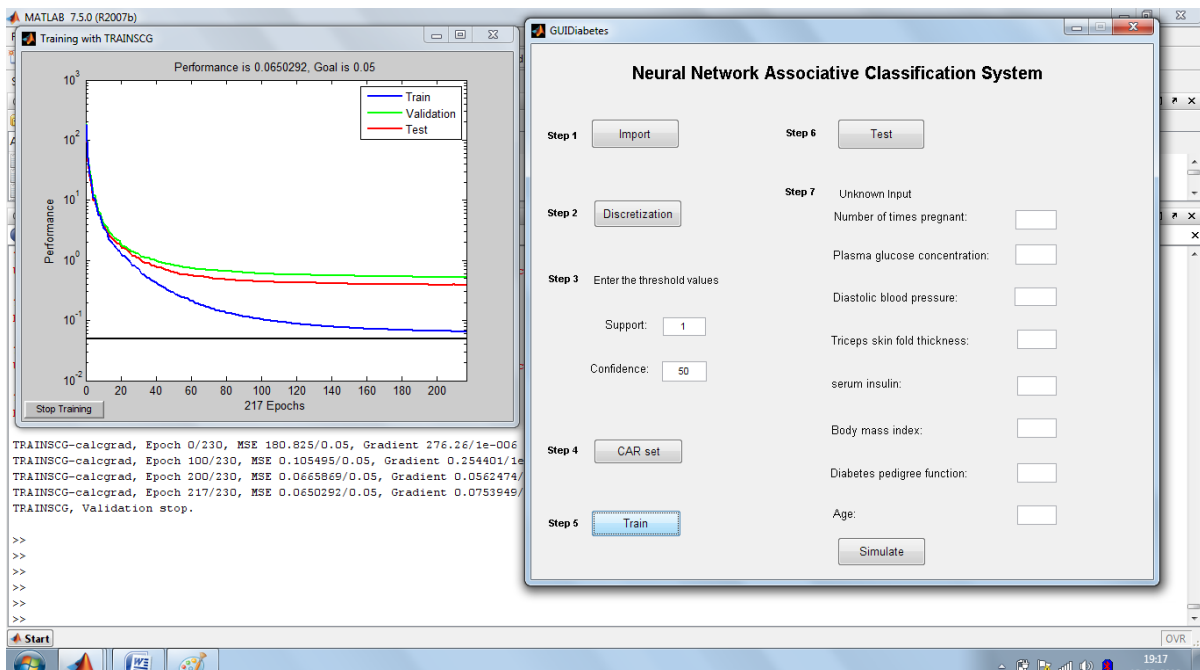


Fig. 7: Training the network

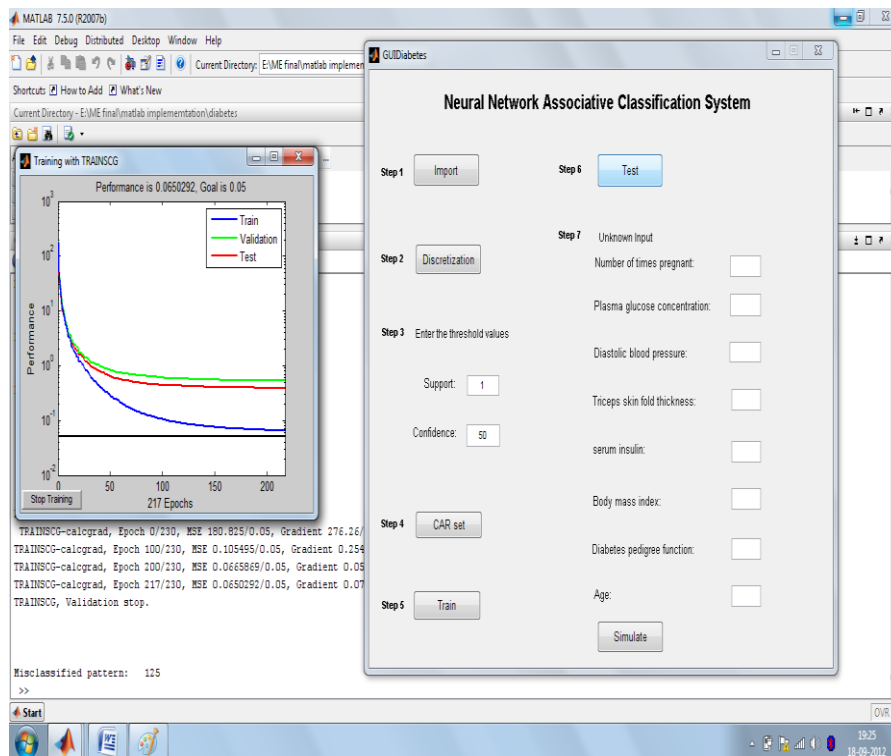


Fig 8: Testing the network

4.3 Result

To assay the performance of classifier for diabetes using Neural Network Associative Classification system we will compare it with Classification Based Associations (CBA) on accuracy. Table 8 shows the accuracy of CBA and Neural network Associative Classification system on accuracy. Class association rules are obtained with minimum support= 1% and minimum confidence = 50%.

Table 8: Comparisons between the CBA system and the Neural Network Associative Classification system on accuracy

Dataset	CBA	Neural network Associative Classification system
Diabetes	74.5	79.236

From the experimental results, classifier for iris plant using Neural Network Associative Classification system will outperform than classifier for diabetes using CBA in accuracy because neural network will learn to adjust weight from the input that is class association rules and will build a more accurate and efficient classifier.

5. Conclusion

Neural network Associative Classification system is used for building accurate and efficient diabetes classifiers in order to improve its accuracy. The three-layer feed-forward back-propagation networks were developed by using Matlab. The optimal values for momentum constant, learning rate and epochs were found to be 0.8, 0.1 and 230 respectively. The structure of networks reflects the knowledge uncovered in the previous discovery phase. The trained network is then used to classify unseen diabetes dataset.

Diabetes classifier using Neural Network Associative Classification system is more accurate than classifier build using CBA as shown in experimental result because weights are adjusted according to the class association rules and best one is used to classify the data.

In future work, we intend to apply predictive apriori algorithm for finding the class association rules instead of applying apriori algorithm.

References

[1]. U. Fayyad, G. Piatetsky-Shapiro, P. Smyth and R. Uthurusamy, "Advances in Knowledge Discovery and Data Mining", MIT Press, Cambridge, Massachusetts, USA, 1996.

- [2]. Mary DeRosa, "Data Mining and data analysis for counterterrorism", Center for Strategic and International Studies, March 2004.
- [3]. Xindong wu, Vipin Kumar, et.al. "Top 10 algorithms in data mining", Knowledge Information System(2008) 14:1-37 DOI 10.1007/s10115-007-0114-2, Springer-Verlag London Limited 2007.
- [4]. R.Agrawal, T.Imielinski, and A.Swami, "Mining Association Rules between sets of items in large databases", SIGMOD, 1993, pp. 207-216.
- [5]. B.Liu, W. Hsu and Y.Ma, "Integrating Classification and Association rule mining", KDD, 1998, pp. no.80-86.
- [6]. Jiawei Han and Micheline Kamber, "Data Mining Concepts and Techniques", 2e Elsevier Publication, 2008.
- [7]. Robert J. Schalkoff, "Artificial Neural networks", McGraw Hill publication, ISE.
- [8]. Chung, Kusiak, "Grouping parts with a neural network", Journal of Manufacturing System, volume 13, Issue 4, 02786125, April 2003, pp.no. 262.
- [9]. Prachitee B. Shekhawat, Sheetal S. Dhande, "A classification technique using associative classification", International journal of computer application(0975-8887) vol. 20-No.5, April 2011, pp.no. 20-28.
- [10]. Cheng Jung Tsai, Chein-I Lee, Wei-Pang Yang, "A discretization algorithm based on Class Attribute Contingency Coefficient", Inf. Sci., 178(3), 2008, pp. no. 714-731.
- [11]. <http://archive.ics.uci.edu/ml/datasets.html>.
- [12]. Xiaoxin Yin, Jiawei Han, "CPAR: Classification based on Predictive Association Rules", In Proc. Of SDM, 2003, pp.no. 331-335.