ISSN No. 0976-5697
Volume 2, No. 1, Jan-Feb 2011
International Journal of Advanced Research in Computer Science
RESEARCH PAPER
Available Online at www.ijarcs.info

A New Discretization and Pattern Selection Method For Classification in Data Mining Using Feedforward Neural Networks

M.Gethsiyal Augusta
Lecturer, Dept of MCA
Sarah Tucker College (Autonomous)
Tirunelveli-627007, South India
augasta@yahoo.com

Dr.T.Kathirvalavakumar*
Head, Dept of computer science
V.H.N.S.N College, ViruduNagar -626001
South India
kathirvalavakumar@yahoo.com

Abstract. This paper proposes a new supervised mean wise discretization algorithm and pattern selection method. A new supervised mean discretization algorithm automates the discretization process based on the mean value of discretizing attribute in each target class. The results obtained using this discretization algorithm show that the discretization scheme generated by the algorithm almost has minimum number of intervals and requires smaller discretization time. A new pattern selection method proposed in this paper is to select the discretized patterns with various features based on pattern disparity for training the feedforward neural network which leads to the improvement in convergence speed and classification accuracy. The efficiency of the proposed algorithm is shown in terms of better discretization scheme and better accuracy of classification by implementing it on six different real data sets.

Keywords: supervised discretization; classification; data mining; pattern selection; backpropagation training algorithm; multilayer feedforward neural network.

I. INTRODUCTION

Many classification algorithms have been developed for classifying real world data sets. However all algorithms can not be applied to the real world classification tasks involving continuous attributes. These continuous attributes to be first discretized. To handle this problem a lot of discretization algorithms have been proposed [1,2,3,4]. Discretization is usually performed prior to the learning process and helps the experts to understand the data more easily and make the learning more accurate and faster [25]. Discretization transforms continuous attributes values into a finite number of intervals and associates with each interval a numerical discrete value. Discretization techniques can be specified in five dimensions such as supervised Vs unsupervised, static Vs dynamic, global Vs local, top-down Vs bottom-up and direct Vs incremental [2]. Many discretization algorithms have been developed in data mining for knowledge discovery. Equal-W and Equal-F are the best examples for unsupervised [5]. In Equal-W method the range of values is simply divided into sub ranges of equal extent and in Equal-F method the range is divided into sub ranges containing equal number of examples. The entropy based and Chi-square based methods are the examples for the supervised procedure [5]. The best examples for the supervised top-down algorithms are Information Entropy Maximization [4], CACC [2] and CAIM [1]. These algorithms generally maintain the highest interdependence between target class and discretized attributes, and attain the best classification accuracy. The famous algorithms in bottom-up methods are chi-merge [6], chi2 [7], modified chi2 [8] and extended chi2 [9]. Wu et al. [10] proposed a dynamic discretization algorithm to enhance the decision accuracy of naive Bayes classifiers. Classification is one of the important functions of data mining. Classification can be performed using different methods such as decision trees [11], Bayesian classification [12], neural networks [13,14,15,16] and genetic algorithms[17]. Among them we select the neural networks as our classification tool as its high tolerance of noisy data, its ability to classify patterns on which they have not been trained and it can be used when there is little knowledge of the relationships between attribute and classes .One of the most common type of neural network is the feedforward backpropagation network. The basic structure of the neural network in this work is a standard three layered feedforward neural network, which consists of an input layer, a hidden layer and an output layer. A backpropagation algorithm with momentum [18] performs learning on this network.

Pattern selection is an active learning strategy to select the most informative patterns for training the network. This is the most important one for obtaining good training set to increase the performance of a neural network in terms of convergence speed and generalization. The level of generalization, i.e., the ability to correctly respond to novel inputs is heavily dependent on the quality of the training data. Much research have been done to improve generalization and to reduce the convergence time. Cohn et al,. [19] have suggested that with careful dynamic selection of training patterns, better generalization performance may be obtained. Vijayakumar and Ogawa [20] have proposed the strategies that allow to dynamically selecting training patterns from a candidate training set in order to reduce the convergence time and to increase the generalization ability of neural networks. For classification problems, Huyser and Horowitz [21] have shown that a network trained on border patterns i.e., the patterns that lie closest to the separating hyper planes generalizes better than a network trained on the same number of examples chosen at random. Wann et al., [22] have used the nearest neighbor criterion to distinguish between typical samples and confusing samples. Many researches [23,24,25,26] have proposed different criteria for selecting training patterns to classify them. Mostly the selection methods are based on K-NN, clustering, confidence measure, Euclidian distance etc.

This paper consists of two phases. In the first phase, a new supervised discretization method is proposed to automatically discretize the continuous attributes of large datasets into discrete intervals and in the second phase a novel pattern selection mechanism is proposed to select the most informative training patterns based on pattern disparity in advance of the training phase from the patterns discretized in the first phase. The proposed discretization algorithm and pattern selection method is aimed at to reduce the training time of neural network and also to improve the accuracy, efficiency and scalability of the classification process. This paper is organized as follows: Section 2 explains the methodology of the proposed discretization process and describes the new discretization algorithm for preprocessing in data mining; Section 3 explains the procedure of the proposed pattern selection mechanism; Section 4 evaluates the performance of the proposed methods using the six real experimental datasets namely iris, Wisconsin breast cancer, Pima Indian diabetes, Ionosphere, Heart and Wave; Section 5 compares the results of the proposed method with other discretization methods in terms of discretization time and classification accuracy.

II. DISCRETIZATION METHOD - MDC

The proposed method finds the mean value of all data in the class k of the continuous attribute to be discretized and splits the data into many intervals by the computed mean value and it is called as a Mean wise Discretization method for Continuous attributes (MDC) as it has been discretizing continuous attributes by mean value. Consider a dataset with N continuous attributes, M examples and S target classes. To classify an example in a dataset using classification algorithm, the existing N continuous attributes should be discretized. k represents an index to a target class where k = 1, 2, ...S. Let max_k, min_k, sum_k and cnt_k respectively represent the maximum, minimum, sum and count of all data values of the discretizing attribute in the class k. The discrete intervals within the range [min_k, max_k] to be generated based on best interval length. Best interval length for all the values of discretizing attribute in class k is computed using the mean $E_k = sum_k / cnt_k$.

The dynamic variable t specifies the value from which the discretization process should be begun. Here t is a dynamic variable and it holds the value with the following condition,

$$\text{When } k = 1, \quad t = min_1$$
$$\text{When } k > 1, \quad t = \begin{cases} min_k & \text{if } max_{k-1} < min_k \\ max_{k-1} & \text{otherwise} \end{cases} \tag{1}$$

Initially t starts with the minimum value of class 1 and consequently it is assigned as either min_k or max_{k-1}, for all $k = 2$ *to* S. Assigning min_k as t for all k classes where $k > 1$ and $max_{k-1} > min_k$ will lead to the generation of redundant intervals. So assigning $t = max_{k-1}$ instead of min_k as avoids the repeated discretization of data in the overlapped area. In order to obtain the good quality for a discretization, finding the best interval length against the continuous valued attribute is considered as the primary vital task. The best interval length l_k for each target class k of a discretizing attribute can be obtained by

$$l_k = |E_k - t| \tag{2}$$

The distance between the mean value and the minimum value of the discretized attribute of each class defines the best interval length for that class. If the best interval length l_k is too small, it raises the discretization time in finding the intervals. In this case the interval length l_k should be increased by finding the value of l_k as $\sqrt{l_k}$ until it becomes greater than *5%* of $(max_k - min_k)$. If the best interval length l_k is too large i.e., closer to total data length $(max_k - min_k)$, the interval length l_k should be reduced by dividing the value of l_k by 2.

Now the number of intervals n for each target class k of discretizing attribute in the discretization scheme is calculated by dividing the total length of data to be discretized in class k by the best interval length l_k of that class. The difference between max_k and the dynamic variable t identifies the total length of data to be discretized in class k. Number of intervals n is defined as follows;

$$n = (max_k - t) / (l_k) \tag{3}$$

First interval namely, d_0 to be kept for the values smaller than min_1 and final interval namely d_m, to be kept for the values greater than max_S for the generalization and to avoid the information loss. The variables l_b and u_b denote the lower bound and the upper bound of an interval. Initially the lower bound value of first interval lb_{11} is min_1. The intervals in the Discretization Scheme *(D)* can be written as,

$$D = \{d_0, d_{k1}, d_{k2}, d_{k3}, ..., d_{ki}, ..., d_{kn}, d_m\} \tag{4}$$

where k varies from $1,2,...,S$,

$$d_0 = values < lb_{11}, d_{k1} = [lb_{k1}, ub_{k1}], d_{k2} = [lb_{k2}, ub_{k2}]$$
$$. \quad . \quad .$$
$$d_{ki} = [lb_{ki}, ub_{ki}]$$
$$. \quad . \quad .$$
$$d_{kn} = [lb_{kn}, ub_{kn}], d_m = values > ub_{Sn}$$

Here $lb_{ki} = ub_{ki-1}$ and $ub_{ki} = lb_{ki} + l_k$.

The steps of the proposed discretization algorithm MDC which requires no sorting procedure follows,

MDC Discretization Algorithm

Input:

Dataset with N continuous Attributes, M Patterns and S target classes.

Begin

1. For each continuous Attribute
   - 1.1 Initialize the first interval as d_0 i.e., values < lb_{11}.
   - 1.2 Let the value t as min_1.
   - 1.3 For each target class k.
     - 1.3.1 Find the maximum value max_k, minimum value min_k and the mean value E_k.
     - 1.3.2 Set the value of t using (1)
     - 1.3.3 Compute the best interval length using (2)
     - 1.3.4 Compute the number of intervals using (3)
     - 1.3.5 Compute n number of intervals using (4)

1.4 Include the interval $[ub_{k-1n}, lb_{k1}]$ if $min_k > max_{k-1}$ to cover all possible values of a continuous attribute for each class k.

1.5 Set the final interval as d_m i.e., *values* $> ub_{Sn}$ where ub_{Sn} is the upper bound value of the last interval.

2. The Discretization Scheme *(D)* for S classes would be

$$D = \{d_0, d_{k1}, d_{k2}, d_{k3}, ..., d_{ki}, ..., d_{kn}, d_m\}$$

where k varies from *1,2,...,S*.

Output:

The Discretization Scheme *D*.

III. PATTERN SELECTION (PS) METHOD

A data which was discretized into many intervals by MDC is converted into binary code using the Thermometer coding scheme [27]. The Thermometer coding scheme uses an n bit code to specify an attribute of the pattern if an attribute is discretized into n discrete intervals. The mostly used random selection method selects the data randomly for training and it may select many redundant or similar patterns. Training the networks with similar patterns reduces the performance of neural networks since the network has not been trained with wide range of patterns. The proposed pattern selection algorithm selects all distinct patterns based on pattern disparity for training the feedforward neural network. *P* represents the set of discretized patterns, A is the number of attributes and *S* is the number of target classes. *n* represents the number of bits in each pattern p_i. *e* represents the number of bits differed between p_i and p_j. The pattern p_{ik} is termed as *distinct* in class *k* when the $e > \eta$ for any pattern p_{jk} where $j=1$ *to* n, $j \neq i$ and $\eta = A / S$ otherwise it is termed as *similar* in class *k*. A threshold value η should be computed to identify the patterns with different features. The threshold value η is considered as number of target classes when division of number of attributes by number of target classes is lesser than the number of target classes otherwise resultant of division. Identify the set of patterns *R*, for training the network by selecting all distinct patterns of *P* and identify the set of patterns *T*, for testing the network by selecting all similar patterns of *P*.

1. Let P be the set of discretized patterns, *A* be the number of attributes and *S* be the number of target classes;
2. If $(A / S) > S$ then $\eta = A / S$ else $\eta = S$;
3. For each attribute *i*
4. For each class *k*
5. Select a pattern p_{ik} from *P* randomly;
6. $R=R+\{p_{ik}\}$; $P=P-\{p_{ik}\}$;
    6.1. For each pattern p_{jk} ,$j\neq i$ of P
        6.1.1. Compare p_{ik} and p_{jk} and find number of differed bits *e*;
        6.1.2. If $e<=\eta$ then $T=T+\{p_{jk}\}$; $P=P-\{p_{jk}\}$;
    6.2. end
7. end

Figure 1. Steps of proposed pattern selection method

The algorithm proposed in Fig. 1 describes the steps for selecting the training patterns and testing patterns of feedforward neural network. The proposed pattern selection algorithm identifies set of most informative patterns from the whole discretized datasets for training the network and considers the remaining patterns as the patterns for testing the network. The advantages of the proposed algorithm are i. Training the network with distinct data helps the network to classify a wide range of testing data. ii. No retraining is necessary since it selects the data in offline. iii. Giving only the informative patterns as an input to the network for training, it guides the network to learn the problem more accurate and faster.

IV. EXPERIMENTAL EVALUATIONS

The proposed algorithm is implemented in JDK1.5 on six well known continuous and mixed mode WEKA's datasets and compared with other discretizaion methods such as Equal-w, Equal-F, Chimerge, Ex-chi2, CACC and CAIM. All experiments were run on a PC with Windows XP operating system, Pentium IV 1.8GHz CPU and 504MB SDRAM memory. Six datasets namely Iris Plants (iris), Ionosphere (iono), Statlog project Heart disease (hea), Pima Indians Diabetes (pid), Wave form (wav) and Wisconsin-breast-cancer (Breastw) are used to test the proposed algorithms. The detailed description of the datasets is shown in Table I.

Table I. Properties of six real datasets

Properties	*Datasets*					
	iris	iono	hea	pid	wav	breastw
# of classes	3	2	2	2	3	2
# of examples	150	351	270	768	5000	699
# of training examples	75	176	135	384	2501	350
# of testing examples	75	175	135	384	2499	349
# of attributes	4	34	13	8	40	9

A. *Results*

Experiments were performed for the proposed MDC algorithm with all the datasets. The MDC algorithm is applied to the entire dataset as the method is global and the results obtained by this algorithm with the six datasets are shown in Table II.

Table II. The results of MDC on six datasets

Criterion	*Datasets*					
	iris	iono	heart	pid	wav	breastw
Mean Number of Intervals	10.5	4.9	6.6	9.75	6.33	7.67
Discretization time (s)	0.01	0.03	0.02	0.02	0.21	0.05

A discretization scheme with very fewer intervals may not only lead to the best quality of discretization scheme, it may lead to decrease in the accuracy of a classifier [2]. The proposed algorithm generates minimum number of intervals, not very few but leads to highest classification accuracy with best discretization time. Here to find the discretization scheme *(D)*, the mean value in each target class *k* of a discretizing attribute is used for all six datasets. For the wave form dataset, the interval length l_k has been increased for an attribute by repeatedly computing the $\sqrt{l_k}$, as it is too small to find the Discretization Scheme. For the ionosphere and heart datasets the interval length l_k has been reduced for an attribute by dividing the $\sqrt{l_k}$ by *2* as it equals the total data length.

The data that was discretized by MDC are converted into binary format using the Thermometer coding scheme[27] and classified with feedforward neural network using backpropagation algorithm.

Table III. The accuracy obtained by BPN on six datasets.

Criter-ion	*Datasets*					
	iris	iono	heart	pid	Wav	**breastw**
Topo-logy	42-3-3	167-5-2	86-5-2	78-5-2	253-5-3	69-3-2
Mse	0.02	0.001	0.01	0.06	0.07	0.001
No. of Epochs	100	92	102	100	100	62
Learning time (s)	0.14	1.11	0.39	0.66	15.3	0.52
Learning rate	0.9	0.1	0.9	0.9	0.1	0.9
Acc (%)	96	93.7	86.7	76.8	79.4	96.6

The training and testing examples are selected using the newly proposed pattern selection method. The average of the results of the 10 experimental runs based on the proposed pattern selection of training and testing examples is calculated for each dataset. The results obtained for ten datasets using multilayer feedforward neural network with backpropagation algorithm (BPN) are shown in Table III. Here the highest classification accuracy is achieved within minimum number of epochs. This algorithm uses the value of momentum as 0.5 but the number of hidden nodes depends on the problem. Normally the data discretized with unsupervised discretization algorithms or with some supervised algorithms requires long training time [1]. But the data selected using the proposed pattern selection method from MDC algorithm (MDC+PS) achieves fast convergence with good accuracy during the classification process using neural network.

B. Comparison of Discretization Schemes

The comparisons of six datasets results with other six discretization schemes are shown in Table IV. The discretization schemes Equal-W and Equal- F are two unsupervised methods, Chi-merge, Extended Chi2, CACC and CAIM are four supervised methods. Table IV shows the number of discrete intervals obtained in this experiment. The main goals of discretization should be to improve the accuracy and efficiency of learning algorithm and the discretization process should be as fast as possible.

From the Table IV, we can see that the generated number of intervals of MDC is comparable with all other discretization algorithms except CAIM. Normally the supervised methods require more execution time since they are considering class related information, but the proposed supervised method MDC requires less discretization time due to its low computational cost.

Table IV. Comparison of MDC with other discretization schemes on six datasets

Discretization Methods	*Mean Number of Intervals*						*Discretization Time (s)*					
	iris	iono	heart	pid	wav	breastw	iris	iono	heart	pid	wav	breastw
Equal-W	4.0	20.0	10.0	14.0	20.0	14.0	0.02	1.72	0.12	0.33	9.06	0.26
Equal-F	4.0	20.0	10.0	14.0	20.0	14.0	0.03	1.84	0.12	0.33	9.33	0.27
MDC	**10.5**	**4.9**	**6.6**	**9.75**	**6.33**	**7.67**	**0.01**	**0.03**	**0.02**	**0.02**	**0.21**	**0.05**
Chi-merge	3.5	21.4	7.8	25.6	28.5	4.6	0.09	4.28	0.39	0.94	64.33	0.66
Ex-chi2	7.5	8.8	2.3	20.0	12.2	3.3	0.11	11.11	1.68	3.23	136.03	1.91
CACC	3.0	4.3	6.4	11.2	18.1	2.0	0.08	3.62	0.22	0.90	61.41	0.58
CAIM	3.0	2.0	2.0	2.0	3.0	2.0	0.08	3.43	0.20	0.80	52.38	0.58

Fig. 2 shows that the discretization time of MDC is smaller than all other methods for all datasets.

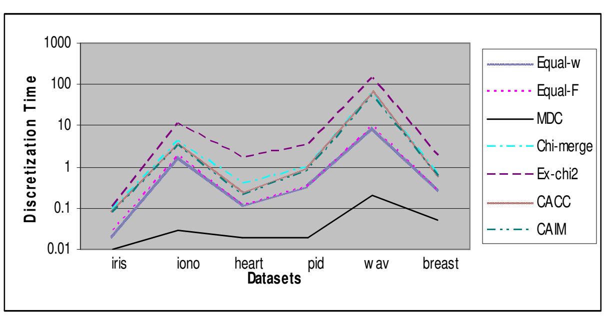

Figure 2. Discretization time comparison of MDC with other methods

The accuracies obtained for the six datasets using neural network (BPN) are compared with the results obtained using c5.0 [2]. The comparison results in the Table V show that the MDC+PS using BPN method reach the highest accuracy among the six discretization algorithms. People often refused to choose the neural network for classifying large datasets, since it requires long training time. But the data discretized by the MDC achieves the highest accuracy with minimum learning time. The comparison of the building time achieved by the BPN using MDC method and by the C5.0 using other discretization methods are also shown in Table V. It shows that the MDC+PS method using BPN requires significantly minimum amount of time for four datasets among six.

Table V. Comparison of the accuracies achieved by the BPNusing MDC+PS and C5.0 using other discretization methods

Algorithm	*Discretization Methods*	*Accuracy (%)* iris	iono	heart	pid	wav	breastw	*Learning Time* iris	iono	heart	pid	wav	breastw
BPN	MDC+PS	96	93.7	86.7	76.8	79.4	96.6	0.14	1.11	0.39	0.66	15.3	0.52
C5.0	Equal-w	91.4	88.0	70.3	70.1	73.2	91.3	1.02	1.04	1.06	1.08	9.25	1.04
	Equal-F	90.8	86.5	72.6	72.8	69.3	90.8	1.02	1.06	1.07	1.08	11.44	1.05
	Chimerge	90.7	87.1	76.5	76.1	69.1	93.0	1.01	1.06	1.06	1.14	12.49	1.04
	Ex-chi2	94.6	88.9	78.4	76.8	76.2	92.3	1.00	1.04	1.05	1.08	10.25	1.02
	CACC	93.5	90.6	78.6	77	79.2	94.1	1.00	1.03	1.05	1.03	8.25	1.02
	CAIM	93.0	90.4	77.1	75.6	78.4	93.8	1.00	1.03	1.05	1.02	9.22	1.02

Moreover the performance of the proposed discretization method MDC+PS is also compared with the performances of a simple unsupervised discretization method Equal-w and a more sophisticated supervised discretization method CAIM on feedforward neural networks using BPN and it is tabulated in Table VI. Here the MDC+PS always achieve the highest classification accuracy for all datasets than Equal-w and CAIM discretization method.

Table VI. Comparison of the classification accuracy and learning time achieved by the BPN using MDC method and by other discretization methods Equal-w and CAIM.

Methods	**Discretiztion Methods**								**No Discretization**	
	MDC+PS		**MDC**		**Equal-w**		**CAIM**		**Continuous Data**	
Datasets	**epochs**	**Accuracy**	**epochs**	**Accuracy**	**Epochs**	**Accuracy**	**Epochs**	**Accuracy**	**Epochs**	**Accuracy**
iris	100	96	100	93.3	100	92	100	94.7	119	**97.3**
iono	92	**93.7**	64	92	249	91.4	167	**93.7**	251	93.1
heart	102	**86.7**	95	85.2	480	76.3	570	81.4	200	78.6
pid	100	**76.8**	100	74.5	100	73.6	100	76.5	100	66.7
wav	100	79.3	100	76.2	100	78.3	100	77.1	100	**81.4**
breastw	62	**96.6**	34	95.1	37	94.3	552	91.9	200	**96.6**

Neural networks are well suited for continuous data, so the performance of MDC+PS is also compared with the performance of continuous data on neural networks. The results show that the MDC+PS discretization method gives better performance for 4 datasets namely ionosphere, heart, pid and breastw.

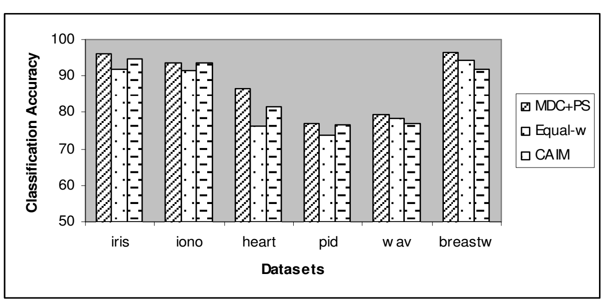

Figure 3. Comparison of the classification accuracy achieved by the BPN using MDC+PS and by other discretization methods Equal-w and CAIM on six datasets.

Fig. 3 compares the classification accuracy of MDC with other discretization methods namely Equal-w and CAIM for six datasets. Table VI also shows the performance of MDC with usual random training patterns selection method. It shows that the data discretized by MDC performs well than the Equal-w method for all datasets except wave and it achieves the highest accuracy for heart and breastw datasets than the CAIM method. The convergence speed of MDC is higher for maximum datasets is also shown in Table 6. For the heart dataset MDC+PS reaches 86.7% accuracy within 102 epochs while CAIM requires 570 epochs. Similarly MDC requires least number of epochs for pid and breastw datasets to achieve the highest classification accuracy than Equal-w and CAIM. Finally the experimental results on data sets show that the proposed algorithm MDC generates the discrete data that results in improved performance of subsequently used learning algorithms when compared to the data generated by other discretization algorithms.

V. CONCLUSIONS

Discretization algorithms have played an important role in data mining and knowledge discovery. In this paper, we proposed the MDC algorithm that handles continuous attributes. The algorithm MDC discussed in Section 2, works with much class labeled data, does not require any user interaction and sorting performs automatic selection of the number of discrete intervals in contrast to some other discretization algorithms. They not only produce a concise summarization of continuous attributes to help the experts to understand the data more easily, but also make learning more accurate and faster. The Section IV shows that the proposed MDC method generates the smallest number of intervals that assumes low computational cost within less amount of time and the discretization time of MDC is smaller than the other discretization methods for maximum datasets. To classify the discretized data using the feedforward neural network with backpropagation algorithm, the most informative training patterns of the feedforward neural network are selected in advance of training phase based on the pattern disparity of discretized patterns using the proposed pattern selection method. Simulation results show that our proposed algorithm MDC with the proposed pattern selection achieves significant improvement in classification accuracy in minimum training time for maximum datasets among the six discretization algorithms. In a nutshell, the MDC algorithm with the proposed pattern selection is very effective and easy to use the supervised discretization algorithm which can be applied to problems that require discretization of large datasets.

VI. REFERENCES

[1] L.A. Kurgan, K.J. Cios, "CAIM Discretization Algorithm", IEEE Transactions on knowledge and Data Engineering, vol. 16, 2004, pp. 45–152.

[2] C.J.Tsai, C.I.Lee, W.P.Yang, "A Discretization algorithm based on Class-Attribute Contingency Coefficient", Information Sciences, 178, 2008, pp. 714 –731.

[3] R.ButterWorth, D.A.Simovici, G.S.Santos, L.O. Machado, "A Greedy Algorithm for supervised discretization", Biomedical Informatics 37, 2004, pp. 285–292.

[4] U.M.Fayyad, K.B.Irani, "Multi-interval discretization of continuous-valued attributes for classification learning", in: Proc. of Thirteenth Int. Conf. on Artificial Intelligence, 1993, pp. 1022–1027.

[5] P.Soman, S.Diwakar, V.Ajay, "Insight into Data Mining", Prentice Hall of India , 2006.

[6] R.Kerber, "ChiMerge: discretization of numeric attributes", in: Proc. of Ninth Int. Conf. on Artificial Intelligence , 1992, pp. 123 –128.

[7] H.Liu, R.Setiono, "Feature selection via discretization", IEEE Transactions on Knowledge and Data Engineering, vol. 9, 1997, pp. 642– 645.

[8] F.Tay, L.Shen, "A modified chi2 algorithm for discretization", IEEE Transactions on Knowledge and Data Engineering, vol. 14, 2002, pp. 666– 670.

[9] C.T.Su, J.H.Hsu, "An extended chi2 algorithm for discretization of real value attributes", IEEE Transactions on Knowledge and Data Engineering vol. 17, 2005, pp. 437– 441.

[10] Q.Wu, D.A.Bell, T.M.McGinnity, G.Prasad, G.Qi, X.Huang, "Improvement of decision accuracy using discretization of continuous attributes", in: Proc. of the Third Int. Conf. on Fuzzy Systems and Knowledge Discovery, Lecture Notes in Computer Science 4223, 2006, pp. 674– 683.

[11] Cohen, S., Rokach, L., Maimon, O.: Decision-tree instance-space decomposition with grouped gain-ratio, Information Sciences 177, 3592– 3612 (2007).

[12] Yager, R.R., An extension of the nave Bayesian Classifier, Information Sciences 176, 577– 588 (2006).

[13] Kaikhah, K., Doddmeti, S.: Discovering trends in large datasets using neural network, Applied Intelligence 29, 51– 60 (2006).

[14] S.Ozekes, O.Osman, "Classification and prediction in data mining with neural networks", Electrical and Electronics Engineering, vol. 3, 2003, pp. 707– 712.

[15] H.Lu, R.Setiono, and H.Liu, "Neurorule: A connectionist approach to data mining", In Proc. of VLDB '95, 1995, pp. 478--489 .

[16] Zhou, Jiang and Chen, "General neural framework for classification rule mining", International Journal of Computers, Systems and Signals, vol. 1, 2000, pp. 154 – 168.

[17] D.Rumelhart, G.Hinton and R.J.Williams "Learning internal representations by error propagation", in Parallel Distributed Processing , MIT Press, Cambridge 1986.

[18] J.Han, M.Kamber, "Data Mining: Concepts and Techniques", Morgan Kaufman ,2001.

[19] D.Cohn, L.Atlas and R.Ladner "Improving generalization with active learning", Machine Learning, vol. 15, 1994, pp. 201– 221.

[20] S.Vijayakumar and H.Ogawa "Improving generalization ability through active learning", IEICE Transactions on Information and Sytems, E82-D(2), 1999, pp. 480– 487.

[21] K.Huyser and A.M.Horowitz, "Generalization in connectionist networks that realize Boolean functions" , in Proc. 1988 Connectionist Models Summer School, Morgan Kaufman, Palo Alto, CA, 1988, pp. 191– 200.

[22] M.Wann, T.Hediger and N.Greenbaun "The influence of training sets on generalization in feed-forward neural network", in Proc. of the Int. Joint Conf. on Neural Networks, vol. 3, 1990, pp. 137– 142.

[23] R.Cheung, I. Lusting, and A.L.Kornhauser, "Relative effectiveness of training set patterns for back propagation", in Proc. of the IEEE Int. Conf. on Neural Networks, vol. 1, 1992, pp. 673– 678.

[24] A.P.Engelbrecht and I.Cloete, "Selective learning using sensitivity analysis", IEEE International Conference on Neural Networks, 1998, pp. 1150 –1155.

[25] B.V.Dasarathy, "Nearest Neighbour (NN) Norms: NN Pattern Classification Techniques", IEEE Computer Society Press, Los Alamitos, CA, 1991.

[26] J. D.Kelly and L.Davis, "A hybrid genetic algorithm for classification", in Proc. of the Twelfth International Joint Conf. On Artificial Intelligence, Sydney, Australia, Morgan Kaufman, 1991, pp. 645-650.

[27] M.Smith, "Neural networks for statistical modeling", Van Nostrand Rainhold, 1993.