

Neural Techniques for Improving the Classification Accuracy of Microarray Data Set using Rough Set Feature Selection Method

Bichitrnanda Patra^{#1}, Sujata Dash^{#2}, B. K. Tripathy^{#3}

¹Department of CSE, KMBB College of Engineering and Technology, Bhubaneswar, Odisha, India

²Department of CSE, GIFT Engineering College, Bhubaneswar, Odisha, India

³VIT-University, Vellore, Tamilnadu, India

Abstract--- Classification, a data mining task is an effective method to classify the data in the process of Knowledge Data Discovery. Classification method algorithms are widely used in medical field to classify the medical data for diagnosis. Feature Selection increases the accuracy of the Classifier because it eliminates irrelevant attributes. This paper analyzes the performance of neural network classifiers with and without feature selection in terms of accuracy and efficiency to build a model on four different datasets. This paper provides rough feature selection scheme, and evaluates the relative performance of four different neural network classification procedures such as Learning Vector Quantisation (LVQ) - LVQ1, LVQ3, optimized-learning-rate LVQ1 (OLVQ1), and The Self-Organizing Map (SOM) incorporating those methods. Experimental results show that the LVQ3 neural classification is an appropriate classification method makes it possible to construct high performance classification models for microarray data.

Keywords-- Data Mining, Rough, Feature Selection, Learning Vector Quantisation, Self-Organizing Map, Classification.

I. INTRODUCTION

Knowledge Data Discovery (KDD) is a process of deriving hidden knowledge from databases. KDD consists of several phases like Data cleaning, Data integration, Data selection, Data transformation, Data mining, Pattern evaluation, Knowledge representation. Data mining is one of the important phases of knowledge data discovery. Data mining is a technique which is used to find new, hidden and useful patterns of knowledge from large databases. There are several data mining functions such as Concept descriptions, Association Rules, Classification, Prediction, Clustering and Sequence discovery to find the useful patterns. Data preprocessing is applied before data mining to improve the quality of the data.

Data preprocessing includes data cleaning, data integration, data transformation and data reduction techniques. Cleaning is used to remove noisy data and missing values. Integration is used to extract data from multiple sources and storing as a single repository. Transformation transforms and

normalizes the data in a consolidated form suitable for mining. Reduction reduces the data by adopting various techniques i.e., aggregating the data, attribute subset selection, dimensionality reduction, numerosity reduction and generation of concept hierarchies. The data reduction is also called as feature selection. Feature selection or attribute selection identifies the relevant attributes which are useful to the data mining task. Applying feature selection with data mining technique improves the quality of the data by removing irrelevant attributes.

Rough set theory was developed by Palwak [1] in the early 1980s and has been used in data analysis, pattern recognition, and data mining and knowledge discovery [2, 3]. Recently, rough set theory has also been employed to select feature subset [4, 5, 1, 6, 7]. In the rough set community, feature selection algorithms are attribute-reduct oriented, that is, finding optimal reduct of condition attributes of a given data set. Two main approaches to finding attribute reducts are recognized as discernibility function-based and attribute dependency-based [8, 1]. These algorithms, however, suffer from intensive computations of either discernibility functions for the former or positive regions for the latter, although some computation efficiency improvement has been made in some new developments.

LVQ was applied successfully to areas such as audio compression, data compression, data transmission, facial recognition, radar signal processing, finance and insurance, production control, sale and marketing, and so on. Keeping all these issues in view, LVQ could be applied to such simple structured data, with higher confidence than that of SOM. One of the most amazing features of LVQ algorithm is that it can take very few vectors to obtain excellent classification results. The idea behind LVQ is to take away codebook vectors from the decision surfaces to clearly demarcate the class borders. We begin with a brief overview of the Rough Set framework in Section 2 and in Section 3 presents the proposed supervised RSAR subset evaluation algorithm. Section 4 describes the learning vector quantization classification algorithms are introduced. The experimental framework and results are

described in Section 5 and 6 respectively. Finally, the conclusion and future work are presented in Section 7.

II. ROUGH SETS

The theory of rough sets begins with the notion of an approximation space, which is a pair $\langle U, A \rangle$, where U is a nonempty set (the universe of discourse), i.e., $U = \{x_1, \dots, x_i, \dots, x_n\}$, and A is a family of attributes, also called knowledge in the universe. V is the value domain of A , and \hat{f} is an information function $\hat{f} : U \times A \rightarrow V$. An approximation space is also called an information system [9].

Any subset P of knowledge A defines an equivalence (also called indiscernibility) relation $IND(P)$ on U

$$IND(P) = \{(x_i, x_j) \in U \times U \mid \forall a \in P, \hat{f}_a(x_i) = \hat{f}_a(x_j)\}$$

If $(x_i, x_j) \in IND(P)$, then x_i and x_j are indiscernible by attributes from P . The partition of U generated by $IND(P)$ is denoted as

$$U/IND(P) = \{[x_i]_P : x_i \in U\}$$

Where $[x_i]_P$ is the equivalence class containing x_i . The elements in $[x_i]_P$ are indiscernible or equivalent with respect to knowledge P . Equivalence classes, also termed as information granules and are used to characterize arbitrary subsets of U . The equivalence classes of $IND(P)$ and the empty set \emptyset are the elementary sets in the approximation space $\langle U, A \rangle$.

Given an arbitrary set $X \subseteq U$, in general, it may not be possible to precisely describe X in $\langle U, A \rangle$. One may characterize X by a pair of lower and upper approximations defined as follows [8]:

$$\underline{P}(X) = \cup \{[x_i]_P \mid [x_i]_P \subseteq X\}$$

$$\overline{P}(X) = \cup \{[x_i]_P \mid [x_i]_P \cap X \neq \emptyset\}$$

That is, the lower approximation $\underline{P}(X)$ is the union of all the elementary sets that are subsets of X , and the upper approximation $\overline{P}(X)$ is the union of all the elementary sets that have a nonempty intersection with X . The tuple $(\underline{P}(X), \overline{P}(X))$ is the representation of an ordinary set X in the approximation space $\langle U, A \rangle$ or is simply called the rough set of X . The lower (respectively, upper) approximation $\underline{P}(X)$ (respectively, $\overline{P}(X)$) is interpreted as the collection of those elements of U that definitely (respectively, possibly) belong to X . The lower approximation is also sometimes called the positive region, denoted as $POS_P(X)$. A set X is said to be definable (or exact) in $\langle U, A \rangle$. If and only if $\underline{P}(X) = \overline{P}(X)$. Otherwise, X is indefinable and termed as a rough set.

III. ROUGH SET ATTRIBUTE REDUCTION (RSAR) SUBSET EVALUATION

The reduction of attributes is achieved by comparing equivalence relations generated by sets of attributes. Attributes are removed so that the reduced set provides the same predictive capability of the decision feature as the original. A reduct is defined as a subset of minimal cardinality

R_{\min} of the conditional attribute set C such that $\gamma_R(D) = \gamma_C(D)$ [10, 11]

The Quick Reduct algorithm given in figure 1, attempts to calculate a reduct without exhaustively generating all possible subsets. It starts off with an empty set and adds in turn, one at a time, those attributes that result in the Quick Reduct(C, D).

C , the set of all conditional features;
 D , the set of decision features.

```

(1) R ← { }
(2) do
(3)   T ← R
(4)   ∀x ∈ (C - R)
(5)     if  $\gamma_{RU(x)}(D) > \gamma_T(D)$ 
(6)       T ← R ∪ {x}
(7)   R ← T
(8) until  $\gamma_R(D) == \gamma_C(D)$ 
(9) return R
    
```

Fig. 1. The Quick Reduct Algorithm

greatest increase in the rough set dependency metric, until this produces its maximum possible value for the dataset.

IV. LVQ CLASSIFICATIONS

LVQ was applied successfully to areas such as audio compression, data compression, data transmission, facial recognition, radar signal processing, finance and insurance, production control, sale and marketing, and so on. Keeping all these issues in view, LVQ could be applied to such simple structured data, with higher confidence than that of SOM. One of the most amazing features of LVQ algorithm is that it can take very few vectors to obtain excellent classification results. The idea behind LVQ is to take away codebook vectors from the decision surfaces to clearly demarcate the class borders [12].

A. The LVQ1

Assume that a number of 'codebook vectors' m_i (free parameter vectors) are placed into the input space to approximate various domains of the input vector x by their quantized values. Usually several codebook vectors are assigned to each class of x values, and x is then decided to belong to the same class to which the nearest m_i belongs.

$$\text{Let } c = \arg \min (||x - m_i||) \quad (1)$$

define the nearest m_i to x , denoted by m_c .

Values for the m_i that approximately minimize the misclassification errors in the above nearest-neighbor classification can be found as asymptotic values in the following learning process. Let $x(t)$ be a sample of input and let the $m_i(t)$ represent sequences of the m_i in the discrete-time domain. Starting with properly defined initial values, the following equations define the basic LVQ1 process [21]:

$$m_c(t+1) = m_c(t) + \alpha(t)[x(t) - m_c(t)]$$

if x and m_c belong to the same class,

$$m_c(t + 1) = m_c(t) \alpha_c(t)[x(t) - m_c(t)]$$

if x and m_c belong to different classes,

$$m_i(t + 1) = m_i(t) \text{ for } i \text{ not in } c.$$

Here $0 < \alpha_c(t) < 1$, and $\alpha_c(t)$ may be constant or decrease monotonically with time. In the above basic LVQ1 it is recommended that α_c should initially be smaller than 0.1; linear decrease in time is used.

B. The LVQ3

The LVQ2 algorithm was based on the idea of differentially shifting the decision borders towards the Bayes limits, while no attention was paid to what might happen to the location of the m_i in the long run if this process were continued. Therefore it seems necessary to include corrections that ensure that the m_i continue approximating the class distributions, at least roughly. Combining these ideas, we now obtain an improved algorithm that may be called LVQ3 [21]:

$$m_i(t + 1) = m_i(t) - \alpha_c(t)[x(t) - m_i(t)],$$

$$m_j(t + 1) = m_j(t) + \alpha_c(t)[x(t) - m_j(t)],$$

where m_i and m_j are the two closest codebook vectors to x , whereby x and m_j belong to the same class, while x and m_i belong to different classes, respectively; furthermore x must fall into the 'window';

$$m_k(t + 1) = m_k(t) + \epsilon \alpha_c(t)[x(t) - m_k(t)] \quad (5)$$

for k in $\{i, j\}$, if x , m_i , and m_j belong to the same class.

C. The optimized-learning-rate LVQ1 (OLVQ1)

The basic LVQ1 algorithm is now modified in such a way that an individual learning rate $\alpha_i(t)$ is assigned to each m_i . We then get the following discrete time learning process. Let c be defined by Eq. (1). Then

$$m_c(t + 1) = m_c(t) + \alpha_c(t)[x(t) - m_c(t)]$$

if x is classified correctly,

$$m_c(t + 1) = m_c(t) - \alpha_c(t)[x(t) - m_c(t)] \quad (6)$$

if the classification of x is incorrect,

$$m_i(t + 1) = m_i(t) \text{ for } i \text{ not in } c.$$

Next we address the problem of whether the $\alpha_i(t)$ can be determined optimally for fastest possible convergence of (6). If we express (6) in the form

$$m_c(t + 1) = [1 - s(t) \alpha_c(t)]m_c(t) + s(t) \alpha_c(t)x(t) \quad (7)$$

where $s(t) = +1$ if the classification is correct and $s(t) = -1$ if the classification is wrong, we first directly see that $m_c(t)$ is statistically independent of $x(t)$. It may also be obvious that the statistical accuracy of the learned codebook vector values is optimal if the effects of the corrections made at different times, when referring to the end of the learning period, are of equal weight. Notice that $m_c(t + 1)$ contains a "trace" from $x(t)$ through the last term in (7), and "traces" from the earlier $x(t')$; $t' = 1, 2, \dots, t-1$ through $m_c(t)$. The (absolute) magnitude of the last "trace" from $x(t)$ is scaled down by the factor $\alpha_c(t)$, and, for instance, the "trace" from $x(t - 1)$ is scaled down by $[1 - s(t) \alpha_c(t)] \alpha_c(t - 1)$.

Now we first stipulate that these two scalings must be identical:

$$\alpha_c(t) = [1 - s(t) \alpha_c(t)] \alpha_c(t - 1) \quad (8)$$

If this condition is then made to hold for all t , by induction it can be shown that the "traces" collected up to time t from the entire earlier x will be scaled down by an equal amount at the end, and thus the "optimal" values of $\alpha_i(t)$ are determined by the recursion

$$\alpha_c(t) = \alpha_c(t - 1) / (1 + s(t) \alpha_c(t - 1)) \quad (9)$$

Any user of the LVQ_PAK can easily become convinced about that (9) really provides for fast convergence. A precaution must be made, however: since $\alpha_c(t)$ can also increase, it is especially important that it does not rise above the value 1; the learning program OLVQ1 is even more restrictive, it never allows any α_i to rise above its initial value[21].

D. The Self-Organizing Map

T. Kohonen introduced the Self-Organizing Map (SOM) [13]. It is an unsupervised learning process, which learns the distribution of a set of patterns without any class information. It has the property of topology preservation. There is a competition among the neurons to be activated or fired. The result is that only one neuron that wins the competition is fired and is called winner-takes all neuron. SOMs may be one-dimensional, two-dimensional or multidimensional, but the most common ones are either one-dimensional or two-dimensional maps.

The number of input connections depends on the number of attributes to be used in the classification. The neuron with weights closest to the input data vector is declared the winner during the training. Then the weights of all of the neurons in the neighborhood of the winning neuron are adjusted by an amount inversely proportional to the distance. It clusters and classifies the data set based on the set of attributes used. The algorithm is summarized as follows [14]:

Step 1- Initialization:

Choose random values for the initial weight vectors $w_j(0)$, the weight vectors being different for $j = 1, 2, \dots, l$ where l is the total number of neurons

Step 2- Sampling:

Draw a sample x from the input space with a certain probability.

Step 3- Similarity Matching:

Find the best matching (winning) neuron $i(x)$ at time steps n by using the minimum distance Euclidean criterion

Step 4- Updating:

Adjust the synaptic weight vector of all neurons by using the update formula where, $\eta(n)$ is the learning rate parameter, and $h_{j, i(x)}(n)$ is the neighborhood function centered around the winning neuron $i(x)$. Both $\eta(n)$ and $h_{j, i(x)}(n)$ are varied dynamically during learning for best results.

Step 5- Continue with Step-2 until no noticeable changes in the feature map are observed.

V. EXPERIMENTS

A. DATASETS

Six used microarray gene expression data sets are chosen for our experiments: Colon tumor, ALL-AML Leukemia, Lung cancer, Prostate_Tumor_GEMS, Brain_Tumor_GEMS and Leukemia_GEMS. The first three data is taken from <http://sdmc.lit.org.sg/GEDatasets/Datasets.html> and other data is taken from <http://www.gems-system.org>. Table I summarize these datasets. We conducted the experiments on these six data sets by applying Rough Set Attribute Reduction (RSAR) Subset Evaluation method for feature reduction and Learning Vector Quantisation (LVQ) - LVQ1, LVQ3, optimized-learning-rate LVQ1 (OLVQ1), and The Self-Organizing Map (SOM) for neural classification of the reduced datasets. We used Weka, a well-known comprehensive toolset for machine learning and data mining [15] as our main experimental platform. We evaluated the performance of feature reduction in Weka environment with four classifiers, using 10-fold Cross Validation.

TABLE I
DATASET INFORMATION

Dataset	# classes	# instances	# attributes
Colon tumor (Train/ Test)	2	40 / 22	2000
Leukemia (Train/ Test)	2	47 / 25	7129
Lung cancer (Train/ Test)	2	109 / 72	12533
Prostate Tumor (Train/ Test)	2	61 / 41	10509
Brain Tumor (Train/ Test)	5	54 / 36	5920
Leukemia_GEMS (Train/ Test)	3	43 / 29	11225

First, a simple method introduced in [16] is used to discretize the domain of each attribute because rough sets methods require discretization input. Any data larger than $\mu + \sigma / 2$ were transformed to state 1; any data between $\mu + \sigma / 2$ and $\mu - \sigma / 2$ were transformed to state 0; any data smaller than $\mu - \sigma / 2$ were transformed to state -1. where σ is standard deviation, μ is mean of a gene. These three states correspond to the over-expression, baseline, and under-expression. Then our method is employed to searching for informative genes for classification.

B. Parameter Settings for RSAR Feature Selection

We used Weka, a well-known comprehensive toolset for machine learning and data mining [17], as our main experimental platform. We evaluated the performance of feature selection methods in Weka environment with an implementation of the Quick Reduct algorithm of rough set

attribute reduction (RSAR) Applicable for use on large gene expression (GE) datasets with numeric continuous data. Evaluate subset using rough set dependency, to return a feature subset giving only the rough set positive region. Feature subset evaluation merit value between 0.0 and 1.0. Not all datasets will reach maximum dependency of 1.0. And Best first may start with the empty set of attributes and search forward with the different parameter as direction : Set the direction of the search as forward, lookup Cache Size : Set the maximum size of the lookup cache of evaluated subsets, at here the default is 1.

After using Rough Set Attribute Reduction (RSAR) subset evaluation and best first search in training dataset, 9 genes are left in colon data set, 3 genes are left in leukemia data set, 13 genes are left in lung cancer data set, 14 genes are left in prostate data set, 36 genes are left in brain tumor data set and 14 genes are left in leukemia_GEMS cancer data set.

C. Comparison of ANN Algorithms with respect to the Parametric Changes

Learning Vector Quantisation (LVQ) - LVQ1. A single BMU (best matching unit) is selected and moved closer or further away from each data vector, per iteration.

*Learning Vector Quantisation (LVQ) - LVQ3.*The same as LVQ1, except only if the classes of the BMU's match, otherwise, the learning rate modified by the epsilon is used on BMU's.

*Learning Vector Quantisation (LVQ) - OLVQ1.*The same as LVQ1, except each codebook vector has its own learning rate. If the BMU has the same class, the individual learning rate is increased, otherwise it is decreased.

Self Organising Map (SOM), Aka Kohonen Feature Map[18]. The SOM algorithm is not intended to be used for classification, this is a version of the SOM that supports supervised learning, as well as unsupervised learning. Class labels can be assigned either dynamically via voting during training, or codebook vector labeling using after training.

To evaluate the performance of the proposed method, the selected feature subsets were evaluated by 10-fold cross validation for the above parameters were changed and correspondingly the correctly classified instances were computed. The intention of these experiments was to bring out the accuracy or overall behavior of the four ANN algorithms, which were shown in Table II without feature selection and with RSAR feature selection shown in Table III.

VI. RESULTS AND COMPARISON

We started experiment by evaluating performance accuracies of different classifiers, LVQ1, LVQ3, OLVQ1 and SOM on four binary and two multi class datasets using 10-fold Cross Validation (CV) without using feature selection algorithms. The results of the 10-fold CV accuracy for the classifiers are shown in table II. After feature selection, the selected feature subsets were evaluated using the above

classification algorithms using 10-fold CV method are shown in Table III.

The experimental results show that the accuracy of microarray data which had feature selection implemented was better than without feature selection datasets. Comparing binary class and multi class datasets, the accuracy of the binary datasets was better than the multi class datasets.

Again, we can observe from Table II and Table III and Graph-1 and Graph-2 that the proposed method effectively increases classification accuracy and selects a smaller number of feature subsets. During the RSAR subset evaluation of the proposed method returns very small sets of genes compared to alternative variable selection methods, while retaining predictive performance. Our method of gene selection will not return sets of genes that are highly correlated, because they are redundant. This method will be useful when considering the design of diagnostic tools, where having a small set of probes is often desirable and to help understand the results from other gene selection approaches that return many genes, so as to understand which ones of those genes have the largest signal to noise ratio and could be used as surrogates for complex processes involving many correlated genes.

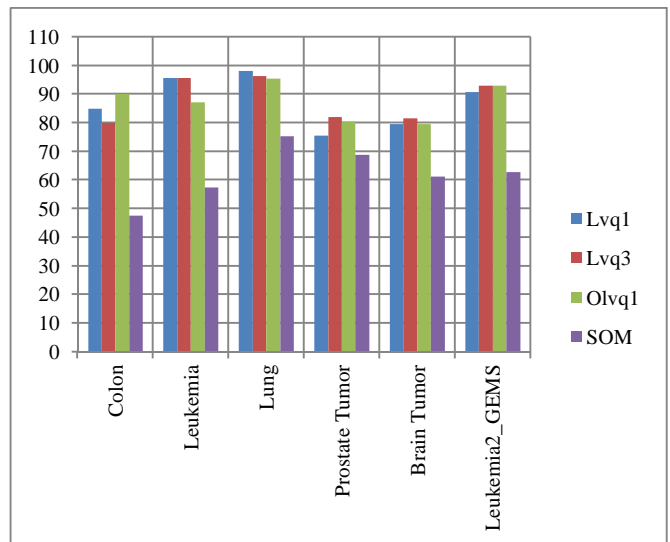
TABLE II
CLASSIFICATION ACCURACY BY USING FOUR ANN CLASSIFIER WITH DIFFERENT PARAMETER SETTINGS BEFORE SUBSET EVALUATION

Datasets	Lvq1	Lvq3	Olvq1	SOM
Colon	85	80	90	47.5
Leukemia	95.7447	95.7447	87.234	57.4468
Lung	98.1651	96.3303	95.4128	75.2294
Prostate Tumor	75.4098	81.9672	80.3279	68.8525
Brain Tumor	79.6296	81.4815	79.6296	61.111
Leukemia2_GEMS	90.6977	93.0233	93.0233	62.7907

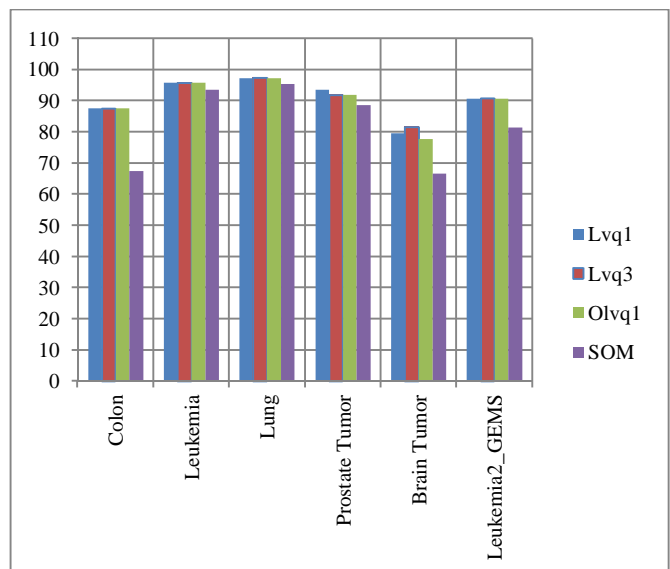
TABLE III

CLASSIFICATION ACCURACY BY USING FOUR ANN CLASSIFIER WITH DIFFERENT PARAMETER SETTINGS AFTER RSAR SUBSET EVALUATION OF TRAINING DATASETS.

Datasets	Lvq1	Lvq3	Olvq1	SOM
Colon	87.5	87.5	87.5	67.5
Leukemia	95.7447	95.7447	95.7447	93.617
Lung	97.2477	97.2477	97.2477	95.4128
Prostate Tumor	93.4426	91.8033	91.8033	88.5246
Brain Tumor	79.6296	81.4815	77.7778	66.6667
Leukemia2_GEMS	90.6977	90.6977	90.6977	81.3953



Graph-1: Classification accuracy by using four ANN classifier with different parameter settings before subset evaluation



Graph-2: Classification accuracy by using four ANN classifier with different parameter settings after RSAR subset evaluation of training datasets.

Based on the comparison of results produced by SOM and LVQ algorithms on the microarray gene expression datasets, LVQ produced better results than SOM, and out of the three LVQ algorithms LVQ3 was the best. The classification accuracy of LVQ with RSAR subset evaluation was found to be in the range of 87.5 to 97.2477 and without subset evaluation was found to be in the range of 75.4098 to 98.

When comparing binary class and multi class datasets, the accuracy of the binary datasets was better than the multi class datasets. For testing data sets of binary class data sets were evaluated using the above classification algorithms using 10-fold CV method are shown in Table IV.

TABLE IV
CLASSIFICATION ACCURACY BY USING FOUR ANN CLASSIFIER WITH
DIFFERENT PARAMETER SETTINGS AFTER RSAR SUBSET EVALUATION OF
BINARY CLASS TESTING DATASETS.

Datasets	Lvq1	Lvq3	Olvq1	SOM
Colon	63.6364	68.1818	59.0909	63.6364
Leukemia	100	100	100	96
Lung	95.8333	95.8333	95.8333	94.4444
Prostate Tumor	87.8049	87.8049	87.8049	92.6829

Table IV shows the classification performances of the four classification algorithms on four microarray data sets, with the highest classification accuracy for each datasets highlighted, LVQ3 achieves the highest classification accuracy in Colon, Leukemia and Lung data sets. On the other hand, SOM achieves highest classification accuracy in Prostate Tumor data set. In general, LVQ3 achieve higher predictive accuracies than the other model and LVQ1, LVQ3 and OLVQ1 achieve highest accuracy for Leukemia dataset. The reliability of this procedure is quite good, although not perfect [19][20].

VII. CONCLUSION AND FUTURE RESEARCH

We conducted an extensive survey in the area of building classification models from microarray data with various ANN classification algorithms. Experimental results show that in most cases, the LVQ3 algorithms delivered classification accuracies equivalent to or better than those on the same data sets reported by other studies. RSAR subset valuation method, which is proved to be an appropriate feature selection method, the learning algorithms are capable of building classification models with high predictive accuracies from microarray data. As the study shows that our feature reduction scheme improves classification accuracies, one question immediately arises: will there be better schemes for the feature selection process for building ANN classification models? Since the number of instances in the studied microarray data is small and the performances of many classification algorithms are sensitive to the number of training data, another interesting question is raised: when comparing predictive performances of various classification algorithms on microarray data, what is the impact of adopting different feature selection methodologies.

REFERENCES

[1] Nguyen, H., Nguyen, S., Some efficient algorithms for rough set methods, IPMU, 1451-1456, 1996
 [2] Lin, T.Y and Cercone, N., Applications of Rough Sets Theory and Data Mining, Kluwer Academic Publishers, 1997.
 [3] Pawlak, Z., Rough Sets: Theoretical Aspects of Reasoning About Data, Kluwer Academic Publishers, 1991.

[4] Han, J., Hu, X., and Lin T. Y., Feature Subset Selection Based on Relative Dependency Between Attributes, 4th International Conference on Rough Sets and Current Trends in Computing, Lecture Notes in Computer Science 3066, pp. 176-185, Springer, 2004.
 [5] Modrzejewski, M., Feature Selection Using Rough Sets Theory, European Conference on Machine Learning, pp.213-226, 1993.
 [6] Quafafou, M. and Boussouf, M., Generalized Rough Sets Based Feature Selection, Intelligent Data Analysis, Vol. 4, pp. 3-17, 2000.
 [7] Sever, H., Raghavan, V., and Johnsten, D. T., The Status of Research on Rough Sets for Knowledge Discovery in Databases, 2nd International Conference on Nonlinear Problems in Aviation and Aerospace, Vol. 2, pp. 673-680, 1998.
 [8] Han, J., Hu, X., and Lin T. Y., A New Computation Model for Rough Set Theory Based on Database Systems, 5th International Conference on Data Warehousing and Knowledge Discovery, Lecture Notes in Computer Science 2737, pp. 381 C 390, 2003.
 [9] Z. Pawlak, *Rough Sets, Theoretical Aspects of Reasoning About Data*. Dordrecht, The Netherlands: Kluwer, 1991.
 [10] Chouchoulas and Q. Shen. Rough set-aided keyword reduction for text categorization. Applied Artificial Intelligence, Vol. 15, No. 9, pp. 843{873. 2001.
 [11] R. Jensen and Q. Shen. Semantics-Preserving Dimensionality Reduction: Rough and Fuzzy-Rough Based Approaches. IEEE Transactions on Knowledge and Data Engineering, Vol. 16, No. 12, pp. 1457{1471. 2004.
 [12] The Learning Vector Quantization Program Package Version 3.1 (April 7, 1995) Prepared by the LVQ Programming Team of the Helsinki University of Technology Laboratory of Computer and Information Science Rakentajanaukio 2 C, SF-02150 Espoo FINLAND
 [13] Self Organizing Maps, Tom Germano, March 23, 1999.
 [14] Face Recognition Using Self-Organizing Map and Principal Component Analysis, Dinesh Kumar, C.S.Rai and Shakti Kumar, 2007.
 [15] Ian H. Witten and Eibe Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, San Francisco, 2 edition, 2005.
 [16] Ding, H.C. Peng, "Minimum Redundancy Feature Selection from Microarray Gene Expression Data", Journal of Bioinformatics and Computational Biology, Vol.3, No.2 , Apr. 2003, pp.185-205.
 [17] Statnikov, C.F. Aliferis, I. Tsamardinos, D. Hardin, S. Levy, "A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis," Bioinformatics, Vol. 21, 2005, No. 5, pp 631-643.
 [18] T. Kohonen, Self-Organizing Maps, Springer, Berlin, 1995.
 [19] Tan, A.C., Gilbert, D., Ensemble machine learning on gene expression data for cancer classification. Applied Bioinformatics, 2003, 2, S75-S83
 [20] S. Dash, B. N. Patra, B. K.Tripathy, "A Hybrid Data Mining Technique for Improving the Classification Accuracy of Microarray Data Set", *International Journal of Information Engineering and Electronic Business(IJIEEB)*, Vol.4, No. April 2012, Page No.43-50.
 [21] T.V. Prasad, Maitrei Kohli, "Vector Quantization of Microarray Gene Expression Data", Proceedings of the World Congress on Engineering 2010 Vol I WCE 2010, June 30 - July 2, 2010, London, U.K