Journal of Theoretical and Applied Information Technology
15th February 2012. Vol. 36 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A COMPARITIVE STUDY OF DIFFERENT DATA MINING CLASSIFICATION TECHNIQUES FOR CANCER MOLECULAR PATTERN DISCOVERY

[1]I. JULIE, [2]DR. E. KIRUBAKARAN
[1]Department of Computer Science, Arignar Anna Government Arts College, Musiri– 621 201
[2]Additional General Manager, BHEL, Trichy – 620 014

Email : [1]julieprakasam@gmail.com

ABSTRACT

The most important application of Microarray for gene expression analysis is used to discover or classify the unknown tissue samples with the help of known tissue samples. Several general purpose Data Mining Classification Techniques have been proposed recently and studied to predict/identify the cancer patterns. In this research work, we have focused and studied a few Classification Techniques such as Support Vector Machine (SVM), Nearest Neighbor Classifier (k-NN), ICS4, Non-Parallel Plane Proximal Classifier (NPPC), NPPC-SVM, Margin-based Feature Elimination-SVM (MFE-SVM). The performance of these classifiers in terms of Minimum Threshold Level to predict/identify the Cancer Pattern, Execution Time, Training Time, Memory Usage and Memory Utilization have been analyzed. This research work has applied these Classification Techniques to 10 publicly available datasets, and compared how these Classification Techniques performed in class prediction of test datasets. From our experimental study, it is observed that for different Cancer Patterns, the threshold levels are different to predict the Cancer Pattern by various Classifiers. It is also revealed that the execution time to predict the cancer patterns are different for different Classifiers. That is overall this work has revealed that although it is obvious that Threshold level based Selection method improves both the memory utilization and execution time but finding the best Classifier for Cancer Prediction is still complicated and the performance and efficiency of Classifier in terms of Execution Time and Memory Utilization is vary in each case.

Key Words: *Microarray, Pattern Recognition, SVM, k-NN, ICS4, NPPC, NPPC-SVM, MFE-SVM.*

1. INTRODUCTION

Bioinformatics is an emerging and rapidly growing field of science. As a consequence of the large amount of data produced in the field of molecular biology, most of the current bioinformatics projects deal with structural and functional aspects of genes and proteins. The data produced by thousands of research teams all over the world are collected and organized in databases specialized for particular subjects. The existence of public databases with billions of data entries requires a robust analytical approach to cataloging and representing this with respect to its biological significance. Therefore, the computational tools are needed to analyze the collected data in the most efficient manner[1,2,3,10].

With the recent development of genomics and proteomics, the molecular diagnostics has appeared as one of the novel tools to diagnose and to predict the cancer patterns. It picks a patient's tissue, serum, and plasma samples and uses DNA chip ie Mass Spectrometry (MS)-based proteomics techniques are used to generate gene/protein expressions of these biological samples. From the generated gene and protein expressions, it could be identified the gene and protein activity patterns in different types of cancerous or precancerous cells. Different cancers have different molecular patterns and the molecular patterns of a normal cell will be different from those of a cancer cell. In modern oncology, clinicians more and more rely on the robust classifications of gene and protein expression patterns to identify cancerous tissues and to find their corresponding biomarkers. However, it is still a challenge for oncologists and computational biologists to robustly classify cancer molecular patterns due to the special characteristics of gene/protein expression data.

In this study, we have mainly focused the various recently proposed Gene Classifiers to

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

measure/predict the accuracy of Cancer Molecular Pattern Identification.

The considered best Gene Classifiers[1,2,3,4,5,6] are Support Vector Machine (SVM), Nearest Neighbor Classifier (k-NN), ICS4, Non-Parallel Plane Proximal Classifier (NPPC), NPPC-SVM, Margin-based Feature Elimination-SVM (MFE-SVM). Our work has implemented the above identified classifiers and thorough comparative study is made. In the following sections, all the above said Classifiers are discussed.

2. DATA MINING CLASSIFIERS

In this section, we are discussing the recently proposed data mining classifiers, which are used for classifying the Cancer Patterns.

2.1 Support Vector Machine (SVM)

Support Vector Machine (SVM)[1] has been a new and promising technique for machine learning. On some applications, it has obtained higher accuracy than Neural Networks. SVM has also been applied to biological problems.

The Support Vector Machine (SVM) is a new and promising technique for data classification and regression. After the development in the past five years, it has become an important topic in machine learning and pattern recognition. Not only it has a better theoretical foundation, practical comparisons have also shown that it is competitive with existing methods such as Neural Networks and decision trees[1,2,3].

Support Vector Machines employ two techniques to deal this case. First it is introduced a soft margin hyperplane which adds a penalty function of violation of constraints to the optimization criterion. Secondly the non-linearly transform the original input space into a higher dimension feature space. Then in this new feature space it is more possible to find a linear optimal separating hyperplane. Given training vectors x_i, i = 1; …. l of length n, and a vector y defined as follows

$$y_i = \begin{cases} 1 & \text{if } x_i \text{ in class } 1 \\ -1 & \text{if } x_i \text{ in class } 2 \end{cases} \quad \text{..... (1)}$$

The Support Vector technique tries to find the separating hyperplane with the largest margin between two classes and it measured along a line perpendicular to the this Hyperplane[1]. For example, in Figure 1, two classes could be fully separated by a dotted line $w^T x + b = 0$. We would like to decide the line with the largest margin. In other words, intuitively we think that the distance between two classes of training data should be as large as possible. That means we want to find a line with parameters w and b such that the distance between $w^T x + b \pm 1$ is maximized.

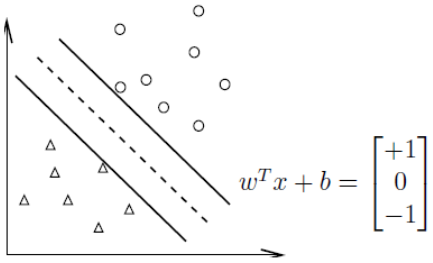

Figure 1. Separating Hyperplane

As the distance between $w^T x + b = \pm 1$ is $2/\|\omega\|$ and maximizing $2/\|\omega\|$ is equivalent to minimizing $\omega^T\omega/2$, we have the following problem

$$\min_{w,b} \quad \frac{1}{2}\omega^T\omega$$
$$y_i((\omega^T x_i)+b) \geq 1, \quad \text{.....(2)}$$
$$i = 1,....,l$$

The constraint $y_i((w^T x_i) + b) \geq 1$ means

$$(\omega^T x_i)+b) \geq 1 \quad \text{if } y_i = 1$$
$$(\omega^T x_i)+b) \leq -1 \quad \text{if } y_i = -1$$

That is, data in the class 1 must be on the right-hand side of $w^T x+b = 0$ while data in the other class must be on the left-hand side. Note that the reason of maximizing the distance between $w^T x + b = \pm 1$ is based on Vapnik's Structural Risk Minimization.

2.2 Nearest Neighbor Classifier k-NN

Among the various methods and techniques of supervised statistical pattern recognition, this Nearest Neighbor[1,2,12] rule does achieve consistently good performance, without *any priori* assumptions, the training examples were drawn. It does involve a training set of both the positive and negative cases. A new sample can be classified by measuring the distance to the nearest training class and then determines the classification of the sample. This *k*-NN classifier can be extended to this concept by taking the *k* nearest points and taking the sign of the majority. This is a common approach to select *k* small and odd to break ties like 1, 3 or 5. The effects of noisy points of training set

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

could be reduced through the Larger *k* values and the *k* is performed with the cross-validation.

There are various approaches/procedures were proposed to improve both the performance and speed of k-NN. The pre-sort is one of the approaches to train the sets. The nearest neighbor rule is very simple, however its computational cost is high. For numerical example, each and every classification does require 60,000 distance calculations between 784 (28x28 pixels) vectors (dimensional one) [12].

This is one of the simplest classification procedures. To classify a query, find the most similar example in D and predict that x has the same label as that example. To carry out this procedure we need to define a similarity measure on expression patterns. This work used the *Pearson correlation* as a measure of similarity. Let

$$k_p(x,y) = \frac{E[(x_i - E[x])(y_i - E[y])]}{\sqrt{Var[x]Var[y]}} \quad .....(3)$$

be the Pearson correlation between two vectors of expression levels. Given a new vector x, the nearest neighbor classification procedure searches for the vector x_i in the training data that maximizes $K_p(x,x_i)$, and l_i , the label of x_i.

This simple non-parametric classification method does not take any global properties of the training set into consideration. However, it is surprisingly effective in many types of classification problems.

2.3 ICS4

Among many classification approaches[1,2,3] using classification rules derived from the decision tree induction may helpful to perform Gene Classification. The general form of rules is presented as

IF condition1 & condition2 & ... & conditionm, THEN a predictive term. The predictive term in a rule refers to a single class label. For useful clinical diagnosis purpose, using those rules we can address issues in understanding the mechanism of a disease and improve the discriminating power of the rules. A *significant rule* is one with a largest coverage which the coverage satisfies a given threshold. For example, the given threshold is *60%*, if one rule's coverage is larger than *60%* then it is called significant rule.

However, the Traditional ensemble method to build and refine the tree committee and derive significant classification rules is still impossible. So, an efficient new incremental decision learning algorithm introduced which uses the skeleton of ITI and accepts the cascading and sharing ensemble method of CS4 to break the constraint of singleton classification rules by producing many significant rules from the committees of decision trees and combine those rules discriminating power to accomplish the prediction process. This algorithm is called as *ICS4* and the procedure is shown below.

```
incremental_update(node, training_example)
{
add_training_example_to_tree(node,
training_example)
{
Add examples to tree using tree revision; }
ensure_best_test(node)
{
 Ensure each node has desired test ; }
sign_class_label_for_test_example(test_example)
 {
    if there are test examples
    for each kth top-ranked tests
   //except the first best test
       force the test to installed at root node
      for remaining nodes
            ensure_best_test(node);
      from each constructed decision tree
        Derive significant classification rules;
  }
}
```

Details of *add_training_example_to_tree* and *ensure_best_test* function are shown in[1]. The third function *sign_class_label_for_test_example* only works at the point when there are test examples or unknown instances that need to be assign a class label. For constructing tree committees, there are two options, construct at the point when we need to perform classification or start from the beginning of tree induction and using incremental manner to construct them. However, at the point when the tree committees are constructed, the top-ranked features used are same for both two strategies because the used examples are no difference. It means that using incremental manner to construct the tree committees is just wasting time and storage. After derived those rules, we use the aggregate score to perform the prediction task. The classification score [1,9] for a specific class, say class *C*, is calculated as

$$Score^c(T) = \sum_{i=1}^{K_c} Coverage(rule^c{}_i) \quad .....(4)$$

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Here, K_c denotes the number of rules in the class *C*, $rule^c_i$ denotes *i*th rules in class *C*, if the score for one class *C* is larger than other classes, then the class label for the instance *T* is assigned as *C*.

2.4 Non-Parallel Plane Proximal Classifier (NPPC) and NPPC-SVM

The Nonparallel Plane Proximal Classifier (NPPC)[3] which is the combine ideas from both Twin

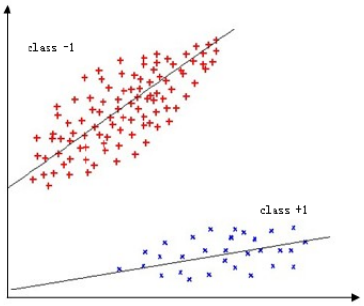

Figure 2. Geometric interpretation of NPPC

Support Vector Machine (TWSVM)[1,3] and Proximal SVM[3].

The NPPC finds two nonparallel hyperplanes such that each plane is clustered around one particular class data, which is shown in the Figure 2. The formulation of NPPC for binary data classification is based on two identical Mean Square Error (MSE) optimization problems which lead to solving two small systems of linear equations in input space. Thus it eliminates the need of any specialized software for solving the quadratic programming problems [3].

$$NPPC_1 \quad \min_{(w_1,b_1,\xi_2)} \ \frac{1}{2}\|Aw_1+e_1b_1\|^2 + C_1e_2^T\xi_2 + \frac{C_2}{2}\xi_2^T\xi_2 \quad ....(5)$$
$$\text{s.t.} \quad -(Bw_1+e_2b_1)+\xi_2 = e_2$$

$$NPPC_2 \quad \min_{(w_2,b_2,\xi_1)} \ \frac{1}{2}\|Bw_2+e_2b_2\|^2 + C_3e_1^T\xi_1 + \frac{C_4}{2}\xi_1^T\xi_1 \quad ....(6)$$
$$\text{s.t.} \quad -(Aw_2+e_1b_2)+\xi_1 = e_1$$

where matrix $A \in R^{m_1 x n}$ represent the data points of class +1 and matrix $B \in R^{m_2 x n}$ represent the data points of class -1 and they contain m1 and m2 training patterns respectively in n dimensional space and m1+m2 = l,w1,w2 $\in R^{m_1}$ are weight vectors and b1,b2 $\in R$ are bias terms of respective planes. C1,C2,C3,C4 > 0 are regularization parameters, e1 $\in R^{m_1}$ and e2 $\in R^{m_2}$ are vectors of ones, $\xi_1 \in R^{m_1}$ and $\xi_2 \in R^{m_2}$ are error variable vectors due to classes +1 and -1 data, respectively. Then two non-parallel hyperplanes $w_1^T x + b_1 = 0$ and $w_2^T x + b_2 = 0$ can be obtained from the solution of $NPPC_1$ and $NPPC_2$. A new data sample $x \in R^n$ is assigned to class +1or -1 depending on which of the two hyperplanes lies closest to the point in terms of perpendicular distance. Finally, the decision function can be written as

$$Class\ k = \underset{k=1,2}{Min}\left|w_k^T x + b_k\right| \quad .....(7)$$

From the literature survey, it is noted that Filters/Wrappers are used to remove noises which will improve the classification accuracy.

To improve the diagnostic accuracy observed that the classification accuracy of a single NPPC is not satisfactory on microarray data by using a small set of informative, Santanu Ghorai and et.al. introduced NPPC ensemble (in place of a single NPPC) with SVM and it is established that this ensemble NPPC-SVM[3] is performing better than single NPPC.

2.5 Margin-based Feature Elimination-SVM (MFE-SVM)

This Margin-based backward Feature Elimination (MFE)[4] is developed for Linear and Nonlinear as well and then it was designed to consider nonlinear kernels. MFE for the nonlinear kernel case experimentally gives both better margin and generalization accuracy. The authors Yaman Aksu and et.al. then present an MFE extension which stepwise (greedily) achieves further gains in margin at small additional computational cost.

This extension solves an SVM optimization problem to maximize the classifier's margin at each feature elimination step, albeit in a very lightweight fashion by optimizing only over a small set of parameters, very similar to a method suggested in [4].

MFE Algorithm Pseudo code for SVMs :

1. *Preprocessing:* Let M be the set of eliminated features, with M=θ initially. First run SVM training on the full space to find a separating hyperplane f(x) =0 (with f parameterized by w,b), with weight norm-squared $L^{-1,0} \equiv \left\|\underline{w}\right\|^2$,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

where I = -1means before eliminating any features and $m_{-1}=0$ is a dummy placeholder index value. For each feature of m, compute $\delta_n^m = y_n x_{n,m} w_m \forall n$. Recall that $g(\underline{x}_n) \equiv y_n f(\underline{x}_n)$ so that δ_n^m is the Δg quantity $\delta_n^{j,m} \equiv (g_n^{j-1,m_{j-1}} - g_n^{j,m})$ whose value is the same at every elimination step for a given pair. Compute $g_n^{-1,0} = y_n b + \sum_{m=1}^{M} \delta_n^m \forall n$. Set I ←0. At elimination step i, perform the following operations.

2. For each m $\notin M$ using recursion, compute $g_n^{i.m} = g_n^{i-1.m_{i-1}} - \delta_n^m \forall n$ determine $N^{i,m} = \min_n g_n^{i,m}$. Determine the candidate feature set $S(i) = \{m \notin M \mid N^{i,m} \geq 0\}$. Note that δ_n^m need not be computed in this step if stored for all m and n during preprocessing. If S(i) is empty then **stop.**
3. For $m \in S(i)$ using recursion, compute $L^{i,m} = L^{i-1,m_{i-1}} - w_m^2$ determine $\gamma^{i,m} = \max_{m \in S(i)} N^{i,m} / \sqrt{L^{i,m}}$
4. In this Step there are three sub-steps
   a. Eliminate feature $m_i \equiv \arg\max_{m \in S(i)} \gamma^{i,m}$, ie $M \rightarrow M \cup \{m_i\}$
   b. Keep for the next iteration only the recursive quantities $\{g_n^{i,m_i} \forall n\}, L^{i,m_i}$ associated with the eliminated feature
   c. i → i+1 and go to step 2

3. PERFORMANCE ANALYSIS AND DISCUSSIONS

In this research work, we have implemented Bio-WEKA based Classification Tool for analyzing the various recently proposed popular Data Mining Classifiers namely Support Vector Machine (SVM), Nearest Neighbor Classifier (k-NN), ICS4, Non-Parallel Plane Proximal Classifier (NPPC), NPPC-SVM, Margin-based Feature Elimination-SVM (MFE-SVM). From our developed tool, all the above said classifiers have been studied thoroughly. For this experimental study, we have used 10 different Cancer Patterns/ Gene Sequences datasets, which are downloaded from National Center for Biotechnology Information (NCBI)[11] website. A few used cancer patterns are Lymphoma Cancer, Breast Cancer, Colon Cancer, Lung Cancer, Melanoma Cancer, Thyroid Cancer, Kidney Cancer, Leukemia Cancer, Pancreatic Cancer, and Endometrial Cancer.

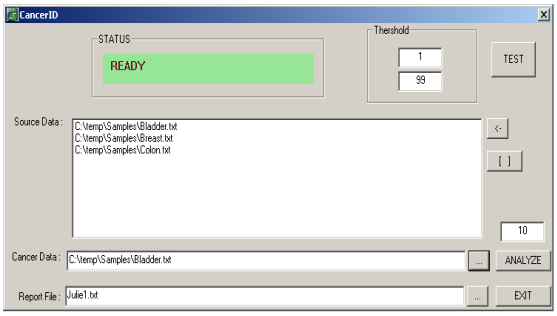

Figure 3. Pattern Integrator and Transformer

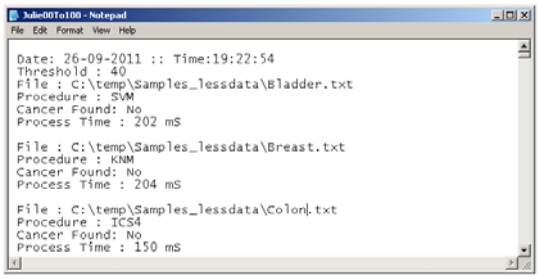

Figure 4. Identification/Prediction Output

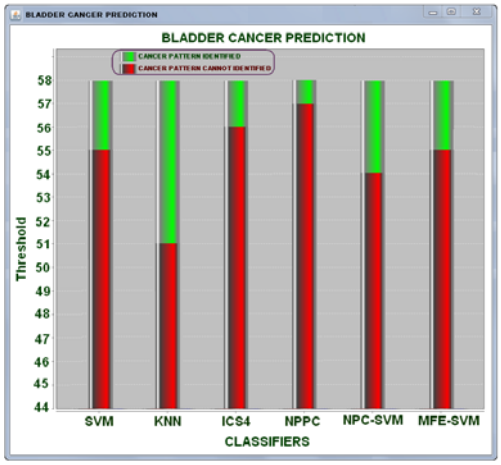

Figure 5. Bladder Cancer Pattern Prediction (Classifiers vs Threshold)

As shown in the Figure 3, we can include Cancer Patters as Sources of Data for comparison. These patterns can be compared and analyzed for different Threshold values from 0 to 100. The prime objective of this work is to analyze the performance of the Classifiers in terms of Execution Time and Threshold Level ranging from 0 to 100.

We analyzed the pattern identification/prediction range of classifiers for

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

different Cancer Patterns and the same is recorded, which is shown in the Figure 4. From the output, we observed that the prediction level (Threshold value) for a Cancer Pattern is different for different Classifiers. It is also noted that the execution time of classifiers are different to predict the cancer pattern.

From the Figure 5, it is observed that for Bladder Cancer Identification/Prediction, kNN predicts this pattern earlier with minimum threshold of 51 and then NPC-SVM identified at 54 and at 55, both SVM and MFE-SVM predicted the same pattern.

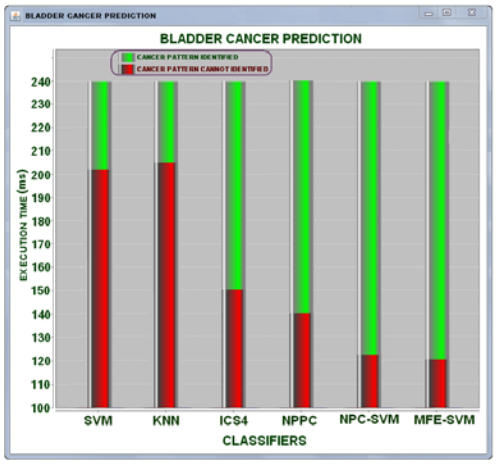

Figure 6. . Bladder Cancer Pattern Prediction (Classifiers vs Execution Time)

ie from this result, it is established that the kNN is the best Classifier to predict the Bladder Cancer Pattern with minimum size of Bladder Gene Sequences. Similarly it performs well for Breast Cancer, which is shown in the Figure 7.

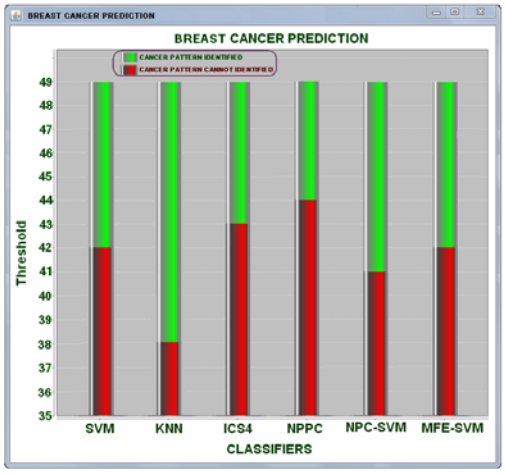

Figure 7. Breast Cancer Pattern Prediction (Classifiers vs Threshold)

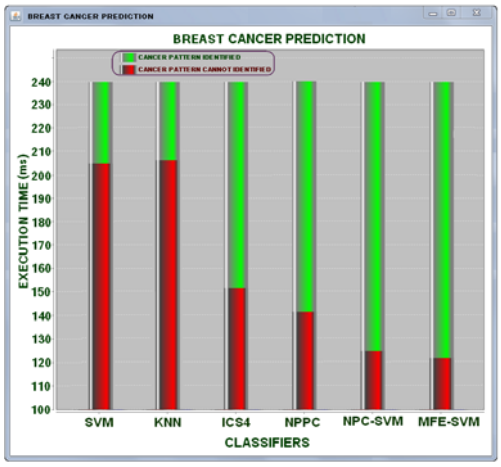

Figure 8. Breast Cancer Pattern Prediction (Classifiers vs Execution Time)

For Colon Cancer Pattern Prediction, like kNN Classifier, the other Classifiers namely SVM, NPPC-SVM and MFE-SVM also predict the Cancer Pattern at the same Threshold Level of 38, which is shown in the Figure 9.

As far as the execution time is concerned to predict the Cancer Pattern, the MFE-SVM always out performs as compared with other identified Classifiers for all types of Cancer Patterns, which is demonstrated from the Figure 6, Figure 8 and Figure 10.

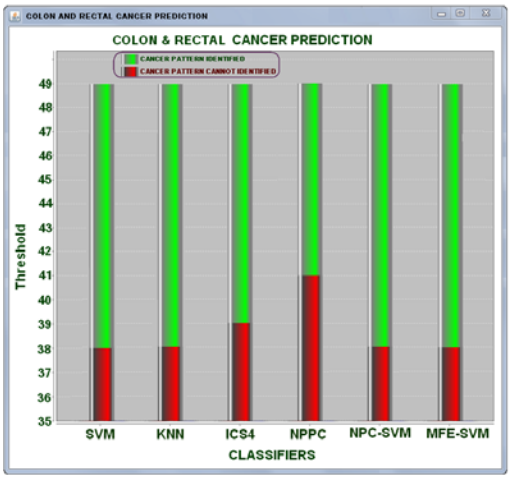

Figure 9. Colon Cancer Pattern Prediction (Classifiers vs Threshold)

Overall this work has revealed that although it is obvious that Threshold level based Selection method improves both the memory utilization and execution time but finding the best Classifier for Cancer Prediction is still complicated and the performance and efficiency of Classifier in terms of Execution Time and Memory Utilization is vary in each case.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

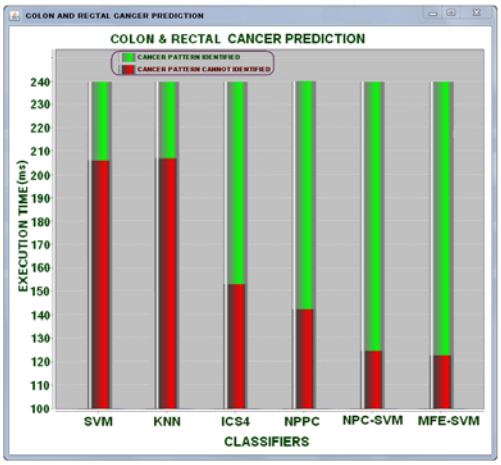

Figure 10. Colon Cancer Pattern Prediction (Classifiers vs Execution Time)

4. CONCLUSION

This research work has focused a few Classification Techniques such as Support Vector Machine (SVM), Nearest Neighbor Classifier (k-NN), ICS4, Non-Parallel Plane Proximal Classifier (NPPC), NPPC-SVM, Margin-based Feature Elimination-SVM (MFE-SVM) and studied thoroughly. The performance of these classifiers in terms of Minimum Threshold Level to predict/identify Cancer Pattern, Execution Time, Training Time, Memory Usage and Memory Utilization have been analyzed. For study, we have applied these Classification Techniques to 10 publicly available datasets, and compared how these Classification methods performed in class prediction of test datasets. From our experimental study, it is observed that for different Cancer Patterns, the threshold levels are different to predict the Cancer Pattern by various Classifiers. It is also revealed that the execution time to predict the cancer pattern is different for different Classifiers.

This research work would like to propose an efficient classifier which will find cancer pattern with minimum threshold level and posses less computational cost as compared with the existing identified classifiers. This would be the future work.

REFERENCES

[1] Minghao Piao, Jong Bum Lee, Khalid E.K. Saeed, and Keun Ho Ryu, "Discovery of Significant Classification Rules from Incrementally Inducted Decision Tree Ensemble for Diagnosis of Disease," International Conference on Advanced Data Mining and Applications, pp. 587–594, Beining, Chinna, 2009.

[2] Hui Ning, Bing Yang, Jun Cui, and Ling Jing, "Detection of Horizontal Gene Transfer in Bacterial Genomes," The Third International Symposium on Optimization and Systems Biology (OSB'09), pp. 229–236, Zhangjiajie, China, September 20–22, 2009.

[3] Santanu Ghorai and et.al, "Cancer Classification from Gene Expression Data by NPPC Ensemble", IEEE Transactions On Computational Biology And Bioinformatics, pp. 1545-5963, 2010.

[4] Yaman Aksu, and David J. Miller, "Margin-Maximizing Feature Elimination Methods for Linear and Nonlinear Kernel-Based Discriminant Functions," IEEE Transactions on Neural Networks, Vol. 21, No. 5, May 2010.

[5] Lin-Kai Luo and et.al., "Improving the Computational Efficiency of Recursive Cluster Elimination for Gene Selection," IEEE/ACM Transactions On Computational Biology And Bioinformatics, Vol. 8, No. 1, January/February 2011.

[6] T.S. Furey, N. Cristianini, N. Duffy, D.W. Bednarski, M. chummer, and D. Haussler, "Support Vector Machine Classification and Validation of Cancer Tissue Samples Using Microarray Expression Data," Bioinformatics, vol. 16, no. 10, pp. 906-914, 2000.

[7] Bogdan Done, Purvesh Khatri, Arina Done, and Sorin Dr_aghici, "Predicting Novel Human Gene Ontology Annotations Using Semantic Analysis," IEEE/ACM Transactions On Computational Biology And Bioinformatics, Vol. 7, No. 1, pp. 91-99, January-March 2010.

[8] Steen Knudsen, Medical Prognosis Institute, "Cancer Diagnostics with DNA Microarrays," A John Wiley & Sons, Inc., Publication, 2006.

[9] Y. Wang, I.V. Tetko, M.A. Hall, E. Frank, A. Facius, K.F.X. Mayer, and H.W. Mewes, "Gene Selection from Microarray Data for Cancer Classification—A Machine Learning Approach," Computational Biology and Chemistry, vol. 29, no. 1, pp. 37-46, 2005.

[10] Avrilia Floratou, Sandeep Tata, and Jignesh M. Patel, "Efficient and Accurate Discovery of Patterns in Sequence Data Sets," IEEE Transactions On Knowledge And Data Engineering, Vol. 23, No. 8, August 2011.

[11] http://www.ncbi.nlm.nih.gov/

[12] http://www.robots.ox.ac.uk/~dclaus/digits/neighbour.htm