Proceedings of the 2007 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB 2007)

Boosting Evolutionary Support Vector Machine for Designing Tumor Classifiers from Microarray Data

Hui-Ling Huang[1], Yi-Hsiung Chen[2], Dwight D. Koeberl[3] and Shinn-Ying Ho[2,4]

[1]Department of Information Management, Jin Wen Institute of Technology, Hsin-Tien, Taipei, Taiwan
[2] Institute of Bioinformatics, National Chiao Tung University, Hsinchu, Taiwan

[3]Department of Pediatrics, Duke University Medical School, Durham, NC, USA

[4]Department of Biological Science and Technology National Chiao Tung University, Hsinchu, Taiwan

***Abstract*—Since there are multiple sets of relevant genes having the same high accuracy in fitting training data called model uncertainty, to identify a small set of informative genes from microarray data for designing an accurate tumor classifier for unknown samples is intractable. Support vector machine (SVM), a supervised machine learning technique, is one of the methods successfully applied to cancer diagnosis problems. This study proposes an SVM-based classifier with automatic feature selection associated with a boosting strategy. The proposed boosting evolutionary support vector machine (named BESVM) hybridizes the advantages of SVM, boosting using a majority-voting ensemble and an intelligent genetic algorithm for gene selection. The merits of the BESVM-based classifier are threefold: 1) a small set of used genes, 2) accurate test classification using leave-one-out cross-validation, and 3) robust performance by avoiding overfitting training data. Five benchmark datasets were used to evaluate the BESVM-based classifier. Simulation results reveal that BESVM performs well having a mean accuracy 94.26% using only 10.1 genes averagely, compared with the existing SVM and non-SVM based classifiers.**

I. INTRODUCTION

Microarray gene expression profiling technology is one of the most important research topics in clinical diagnosis of disease. The practical applications of microarray gene expression profiles include management of cancer and infectious diseases [1]. The normal cells can evolve into malignant cancer cells through a series of mutation in genes that control the cell cycle [2]. However, to identify such an optimal subset from thousands of genes is intractable, which plays a crucial role when classifying multiple-class genes express models from tumor samples. In addition, due to high degree of freedom in the search space, it may occur that there are multiple sets of relevant genes having the same high accuracy in fitting the training data that is so called model uncertainty. How to design an accurate tumor classifier with automatic gene selection and consideration of model uncertainty from microarray gene expression data is investigated in this paper.

Genetic algorithm (GA) [3] is a randomized search and optimization technique that simulates the natural evolution by an iterative computation process. GA can consider multiple interacting attributes simultaneously rather than considering a single attribute at a time. Furthermore, GA is capable of searching for optimal or near-optimal solutions to optimization problems with complex and large search spaces. A number of GA-based gene selection schemes have been used in microarray data analysis. Li *et al*. (2001) [4] proposed a hybrid method of GA-based gene selection and *k*-nearest neighbor classifier to assess the importance of genes for classification. Ooi and Tan (2003) [5] proposed an efficient hybrid approach based on GA and maximum likelihood classification.

Support vector machine (SVM) [6], a supervised machine learning technique, is one of the methods successfully applied to cancer diagnosis problems in the previous studies [7]-[12]. To build an efficient and effective model for classification, it is indicated that SVM performs better than some existing classification algorithms [8]. Statnikov *et al.* [13] investigated classification algorithms which can handle multiple classes and a large number of variables, and compared multi-category SVM to neural networks and k-nearest neighbor classifiers. The results indicate that the multi-category SVM is the most effective classifier for tumor classification.

To cope with multiple sets of relevant features of model uncertainty, some useful approaches have been proposed, such as boosting algorithms [14], [15]. Li and Yang [16] used a model averaging approach to classification of microarray data by averaging over multiple single-gene models. Yeung *et al*. [17] presented a Bayesian model averaging approach as a multivariate feature selection method for multi-class microarray data.

The intelligent genetic algorithm (IGA) is one customized version of the intelligent evolutionary algorithm [18] for solving specific problems. Ho *et al*. [19] proposed an interpretable gene expression classifier using IGA for microarray data analysis. Huang *et al*. [20] proposed an IGA-based classifier by selecting a minimal number of informative genes. The automatic gene selection and parameter tuning are simultaneously optimized by IGA, which can advance the classification performance and is beneficial to factor analysis from a large number of given features. Some of the IEA-based classifier design methods can refer the studies [21]-[23].

This study proposes a boosting evolutionary SVM (name BESVM) based classifier for tumor classification by majority-voting ensemble. The merits of the BESVM classifier hybridizing the advantages of boosting, IGA and

SVM are threefold: 1) automatic gene selection, 2) consideration of model uncertainty, and 3) achieving accurate and robust prediction of unknown samples using the selected features. The majority-voting ensemble classifier [24] is one of the simplest ensemble forms that can combine the outputs of multiple classifiers. By different weighting schemes, they can simply choose the predicted class by plurality from a classifier pool [25], [26]. The possible intervals of C and γ with grid space provide each SVM with parameters (C, γ) as an independent SVM classifier. A threshold strategy in selection with SVMs is applied after the SVM models have been established.

The effectiveness of BESVM is evaluated by designing an accurate tumor classifier with automatic gene selection and consideration of model uncertainty from microarray gene expression data. Five benchmark datasets were used to evaluate the BESVM-based classifier. Simulation results reveal that BESVM performs well having a mean accuracy 94.26% using only 10.1 genes averagely, compared with the existing SVM and non-SVM based classifiers.

II. Support Vector Machine

SVM is a very popular method to deal with classification, prediction, and regression problems. Various SVMs introduced by Vapnik and other co-workers [6], [27] are powerful classifiers. For the binary SVM, the training data consist of n pairs (x_1, y_1), (x_2, y_2),…, (x_n, y_n), with $x_i \in \Re^m$ and $y_i \in \{-1,1\}$, $i = 1, 2, \ldots, n$. The standard SVM formulation is as follows:

$$\min_{w,b,\xi} \ \tfrac{1}{2} w^T w + C\sum_{i=1}^{n} \xi_i \quad \text{subject to} \qquad (1)$$
$$y_i(w^T \phi(x_i)+b) \geq 1-\xi_i,\ \xi_i \geq 0,\ i = 1,\ldots, n,$$

where $w \in \Re^m$ is a vector of weights of training instances; b is a constant; C is a real-valued tradeoff (cost) parameter; ξ_i is a penalty parameter; and ϕ is to map x_i into a higher dimensional space. The SVM of (1) is called a linear kernel SVM when $\phi(x_i)=x_i$. The SVM finds a linear separating hyperplane with the maximal margin in the higher dimensional space. C > 0 is the penalty parameter of error term. The SVM of (1) is called a nonlinear SVM when ϕ maps x_i into a higher dimensional space.

For the nonlinear SVM, the value of variable w can be vary large or even infinite, so it is very difficult to solve using (1). The general method is to use the following dual formulation:

$$\min_{\alpha} \ \frac{1}{2}\alpha^T Q\alpha - e^T\alpha \quad \text{subject to} \qquad (2)$$
$$y^T\alpha=0,\ 0\leq\alpha_i\leq C,\ i = 1,\ldots, n,$$

where e is the vector of all ones, $C>0$ is the supper bound, Q is an $n\times n$ positive semidefinite matrix, $Q_{ij} \equiv y_i\, y_j\, K(x_i\, x_j)$, and $K(x_i,\ x_j) \equiv \phi(x_i)^T\phi(x_j)$ is a kernel function. Some commonly-used kernel functions are: $e^{-\gamma\|x_i-x_j\|^2}$ (Radial basis function), $(x_i^T x_j/\gamma+\delta)^d$ (Polynomial), and $\tanh(\gamma x_i^T x_j+\delta)$ (sigmoid), where γ, d and δ are kernel parameters. The number of variables in (2) is the size n of the training dataset which is smaller than the dimensionality of $\phi(x)$.

Chang and Lin [28] develop a software tool LIBSVM (Library for Support Vector Machine) for support vector classification, regression and distribution estimation. LIBSVM uses the "one-against-one" approach [29] for multiclass classification. In the one-against-one approach, $k(k-1)/2$ classifiers are established where k is the number of classes. The classifiers between each pair of k classes are optimized using the following dual formulation:

$$\min_{i,j} \ \frac{1}{2}(w^{i,j})^T w^{i,j} + C\sum_t \xi_t^{ij} \quad \text{subject to} \qquad (3)$$
$$(w^{ij})^T \phi(t) + b^{ij} \geq 1-\xi_t^{ij}, \quad \text{if } y_t = i$$
$$(w^{ij})^T \phi(t) + b^{ij} \leq -1+\xi_t^{ij}, \quad \text{if } y_t = j,\ \xi_t^{ij} \geq 0.$$

After solving the optimization problem using (3), $k(k-1)/2$ decision functions can be obtained. To predict a class label of a given instance x, the prediction for each of the $k(k-1)/2$ classifiers is calculated using a voting strategy. If there is a class, say j, that receives the largest number of votes, the instance x is assigned to class j, where a tie is broken randomly. One advantage of using this method is that each classifier is easy to train since only the binary SVM is needed. Another approach to multiclass classification is called "one-against-all". In this approach, k models of SVM are established. For each class j, the SVM is trained using all the instances in the class j as positives and the rest of instances as negatives. Previous research has shown that one-against-one outperforms one-against-all for multiclass classification [30].

III. Intelligent Genetic Algorithm

The used IGA to optimize the parameters in S using the fitness function F(S), defined in (4), is given as follows:

Step 1: Initialization: Randomly generate an initial population with N_{pop} feasible individuals where each gene g_i is unique in a GA-chromosome.

Step 2: Evaluation: Evaluate fitness values of all individuals in the population. Let I_{best} be the best individual in the population.

Step 3: Use the simple ranking selection that replaces the worst $P_s \cdot N_{pop}$ individuals with the best $P_s \cdot N_{pop}$ individuals to form a new population, where P_s is a selection probability.

Step 4: Randomly select $P_c \cdot N_{pop}$ individuals including I_{best}, where P_c is a crossover probability. Perform intelligent crossover operations for all selected pairs of parents.

Step 5: Apply a conventional bit-inverse mutation operator to the population using a mutation probability P_m. To prevent the best fitness value from deteriorating, mutation is not applied to the best individual.

Step 6: Termination test: If a pre-specified termination condition is satisfied, stop the algorithm. Otherwise,

go to Step 2.

IV. THE PROPOSED BESVM

The two-level cross-validation achieving accurate and robust prediction of unknown samples using the selected features is applied, using leave-one-out cross validation (LOOCV) in the outer loop and a stratified 10-fold cross-validation (10-CV) in the inner loop [13]. The inner loop is used to determine the best performance of the boosted SVM models on the validation data. The outer loop is used as unknown samples for estimating the performance of the BESVM classifier.

A. Chromosome representation

Let S be the set of parameters $\{t_1,\ldots, t_l, g_1,\ldots, g_l\}$ to be optimized. The control GA-genes t_i are binary variables where the constant l is pre-defined by designers. The parametric genes $g_i \in [1, m]$ are serial numbers of genes in the microarray data. The variable t_i is used to determine whether the corresponding gene of g_i is selected or not. The advantage of using control genes rather than each parametric gene has an equal probability to be evaluated that is beneficial to the used IGA. All the parameters are encoded into a chromosome, as shown in Fig. 1.

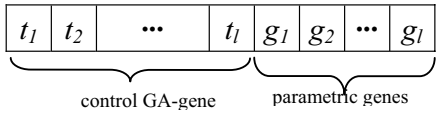

Fig. 1. Chromosome representation.

B. Fitness Function

Fitness value guides IGA to choose offspring for the next generation from the current parents. For achieving the two objectives, maximizing classification accuracy $R(S)$ and minimizing the number $G(S)$ of relevant genes, the fitness function $F(S)$ is a weighted sum with a weight w as follows:

$$\max\ F(\mathrm{S}) = R(S) - wG(S). \tag{4}$$

$G(S)$ is the sum of the values of t_i and $R(S)$ is the accuracy of the ensemble classifier using these $G(S)$ genes. The penalty term $wG(S)$ is to further minimize $G(S)$ while maximizing $R(S)$. The accuracy $R(S)=(RT(S)+RV(S))/2$ where $RT(S)$ is training accuracy and $RV(S)$ is validation accuracy. Therefore, the used fitness function in the following simulations is as follows:

$$\max F(\mathrm{S}) = (RT(S)+ RV(S))/2 - wG(S). \tag{5}$$

C. The used SVM model

The formulation in Section II can take nonlinearly separable cases into account by letting C be finite values. SVM has shown good performance in data classification that depends on tuning of several parameters. The parameters affect the generalization ability. The basic approach to SVM classification may be extended to allow for nonlinear decision surfaces. For this, the input data are mapped into a high dimensional space through a nonlinear mapping function which has effect of spreading the distribution of the data points in a way that facilitates the fitting of a linear hyperplane. The classification decision function is as follows:

$$\mathrm{sgn}(\sum_{i=1}^{n} \alpha_i y_i K(x_i, x) + b) \tag{6}$$

where α_i, $i = 1,\ldots,n$, are Lagrange multipliers. The magnitude of α_i is determined by the parameter C [30]. The RBF kernel function is used:

$$k(x_i\text{-}x)= e^{-\gamma \|x_i - x\|^2} \tag{7}$$

where γ is the parameter controlling the width of the Gaussian kernel. Therefore, tuning the cost parameter C and kernel parameter γ is necessary to solve the classification problems.

The SVM with different values of C and γ is treated as an independent classifier. The 10-fold cross-validation test provides a bias-free estimate of the accuracy at a much reduced computational cost, and is considered an acceptable test to evaluate prediction performance of an algorithm [31]. IGA selects an optimal set S of relevant features for all the input data. To estimate the grid space of the SVM models, the validation accuracy is adopted. For obtaining the kth set of models which can be used to estimate $RT(S)$, a threshold value t is specified that the top-rank t% of the SVM models are selected according to the training accuracy. Namely, there are t% SVM classifiers SVM^k having T SVMs in each fold of 10-CV in the classifiers pool.

D. Majority voting

The majority-voting ensemble [24] is adopted in BESVM to design an accurate tumor classifier with automatic gene selection and consideration of model uncertainty. Based on the majority-voting ensemble for classification, generation, selection and combination can be conceived as outlined in Fig. 2. The majority-voting ensemble classifier is one of the simplest ensemble forms that can combine the outputs of multiple classifiers. By different weighting schemes, they can simply choose the predicted class by plurality from a classifier pool [20], [30]. Assuming that the errors made by the classifiers are not highly correlated, the samples that are not accurately classified by one classifier have a good chance to be correctly classified by a plurality of the other classifiers.

A combination of several SVMs will advance classification accuracy. Majority voting [32] is the simplest method for utilizing the kth set of SVM models SVM^k. Let f_j (j=1, …, T) be a decision class of the jth SVM in SVM^k where $f_j=s \in \{1, 2, \ldots, Ca\}$ and Ca is the total number of classes. Then, let N_s be the number of SVMs whose decision class is class s. Then, the final decision class $f_{mv}(x)$ of SVM^k for a given sample x is determined by

$$f_{mv}(x) = \arg\max_s N_s . \tag{8}$$

The training accuracy $RT_k(S)$ and validation accuracy $RV_k(S)$ of SVM^k are estimated using the majority voting (8).

E. The BESVM classifier

1. Leave one out form N samples of the used dataset.
2. Evolve a set of parameters S using IGA for gene selection through the step 3.
3. Establish a best ensemble classifier SVM^{best} from SVM^k, k=1, …, 10. To establish SVM^k, perform the following steps:
   - 3-a) Split the N-1 samples into 10 folds.
   - 3-b) Evaluate the kth fold where k=1, …, 10 as follows:
     - b-1) Train SVM models with the remaining 9 folds.
     - b-2) Generate the set of SVM models using a grid space (C, γ).
     - b-3) Select the top-rank t% (=T) SVM models. Combine the selected T SVMs as an ensemble classifier SVM^k.
     - b-4) Based on the major-voting strategy, compute validation accuracy using the kth fold.
   - 3-c) Choose the best ensemble classifier SVM^{best} in terms of the validation accuracy.
4. Classify the test sample using SVM^{best} with the major-voting strategy.
5. Calculate the accuracy of LOOCV and mean number of selected genes.

V. EXPERIMENTS

A. Data sets

The proposed BESVM is evaluated using five datasets which are often used in recent literature on the classification problem in analyzing gene expression data. The datasets are described in Table I. The five multicategory datasets are available by download from http://www.gems-systems.org for non-commercial use. The five datasets have 2-5 distinct diagnostic categories, 50-102 patients and 5327-11225 genes, after the data preparatory steps [13]. A simple rescaling of gene expression values to [–1, 1] is performed to utilize SVM.

B. Evaluating the BESVM classifier

The BESVM classifier was implemented using VC++ 6.0 on a PC. The parameters of IGA are as follows: population size N_{pop} = 20, crossover rate P_c = 0.8, truncation rate P_s = 0.2, and mutation rate P_m = 0.2. In the set S of parameters, let the number of control GA-genes be l=15Ca and t=20. In this study we extended the ranges of SVM parameters: cost C ={0.0001×2^d, 0.001×2^d, 0.01×2^d, 0.1×2^d, 1×2^d, 10×2^d, 100×2^d} and γ ={0.0001×2^d, 0.001×2^d, 0.01×2^d, 0. 1×2^d, 1×2^d}, d=0, 1, 2, 3. There are 28×20 grid points of C and γ with grid space. The stopping condition is to use 100 generations of IGA.

For the proposed method, the classification accuracy for each dataset is calculated from results of outer loop LOOCV. The performance comparison is shown in Table II in terms of accuracy and number of used genes. Table II shows the high performance of BESVM where the mean accuracy is 94.26% using only 10.1 genes averagely. Compared with the result of MC-SVM in [13], the mean accuracy is 91.42% without using gene selection. The results reveal that the boosting strategy and the gene selection using IGA are useful to design the SVM-based classifiers in advancing accuracy and number of used genes.

C. Evaluating the boosting strategy

To further evaluate the boosting strategy without interference of the IGA-based gene selection, a prespecified number of genes were selected by the Wilcoxon rank sum test [33] as a non-parametric feature pre-selection method where $G(S)$= 10, 20, …, 100. Three SVM-based classifiers were studied for comparison: 1) 10-CV with major-voting and t=20, denoted as 10CV-Top20%, 2) 10-CV without major-voting using the best one of the 10 SVM models, denoted as 10CV-Best, and 3) no cross-validation is used, denoted as non-CV. The results are shown in Fig. 3. The results of Fig. 3 show that the 10-CV with the boosting strategy is effective on average. The additional cost is the computation time and space for the extra SVM classifiers, which is worth doing.

An advantage of the boosting strategy is its robustness of classification results. Fig. 4 shows the results of the three SVM-based classifiers using a box plot presentation. It is shown that 10-CV with major-voting (10CV-Top20%) is the most robust and accurate classifiers. In addition, the10-CV method is able to cope with overfitting problems.

VI. CONCLUSIONS

The proposed BESVM consisting of SVM, boosting and IGA has been shown effective for designing tumor classifiers from microarray data. The number of genes is usually much greater than the number of tissue samples available, and only a small subset of the genes is relevant in distinguishing different classes. Considering this characteristic of microarray data, the classifier design should avoid overfitting the training data in selecting a small set of genes by maximizing the performance of independent tests.

It is essential to select a minimal number of relevant genes from microarray data while maximizing classification accuracy of independent tests for the development of inexpensive diagnostic tests. After computer simulation using five benchmark datasets, it reveals that BESVM could obtain not only higher classification accuracy but also a smaller number of relevant genes than the existing methods. In addition, the IGA-based gene selection method is an efficient method in designing classifiers for analyses of microarray data.

This study has shown the individual abilities of SVM with boosting and IGA for gene selection. To our knowledge, the BESVM method has the best classification performance in terms of the number of used genes (10.08 genes on average) and test accuracy (94.26% using leave-one-out cross-validation).

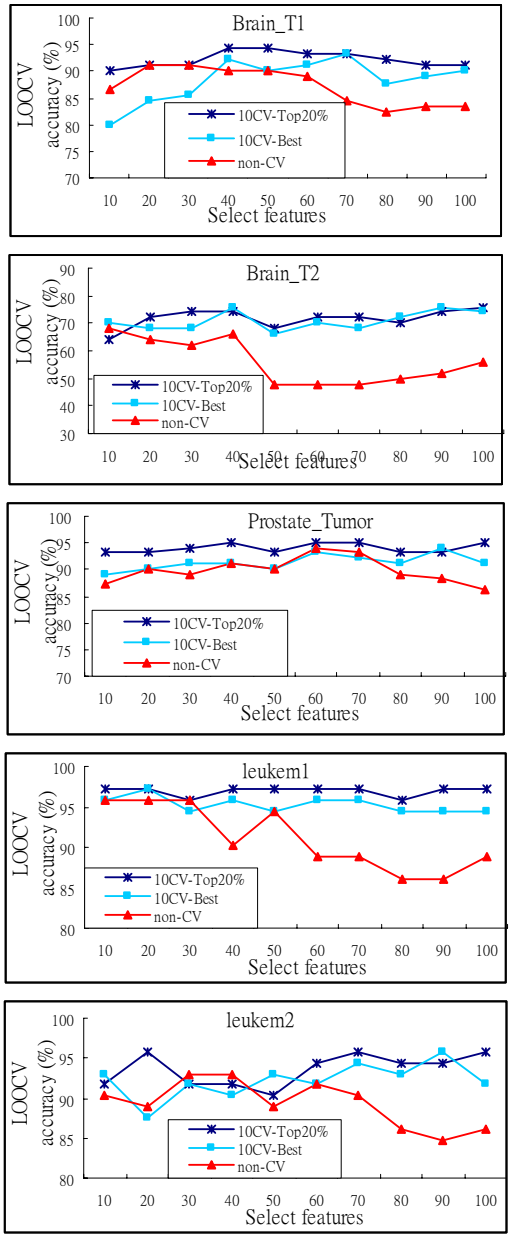

Fig. 3. Performance comparisons of three SVM-based classifiers for evaluating the boosting strategy using major-voting ensemble on five datasets in terms of LOOCV accuracy and number of used genes.

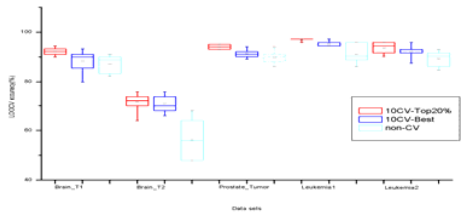

Fig. 4. The results of three inner procedures using a box plot presentation.

REFERENCES

[1] P. Fortina, et al., "Molecular diagnostics: hurdles for clinical implementation," *Trends Mol Med*. 8(6), pp. 264-6, Jun. 2002.

[2] A. Ben-dor, et al., "Tissue classification with gene expression profiles," *In Proceeding of the Fourth International Conference on Computational Molecular Biology (RECOMB2000)*, ACM Press, New York, 2000.

[3] D.E. Goldberg, *Genetic Algorithms in search, Optimization and Machine Learning*: Addison-Wesley Publishing Company, 1989.

[4] L. Li, R. Clarice, T.A. Darden and L.G. Pedersen, "Gene selection for sample classification based on gene expression data: study of sensitivity to choice of parameters of the GA/KNN method," *Bioinformatics*, 17, pp. 1131-1142, 2001.

[5] C.H. Ooi and P. Tan, "Genetic algorithms applied to multi-class prediction for the analysis of gene expression data," *Bioinformatics*, 19, pp. 37-44, 2003.

[6] V. N. Vapnik, *Statistical Learning Theory*. New York, Wiley, 1998.

[7] T.S. Furey, et al., "Support vector machine classification and validation of cancer tissue samples using microarray expression data," *Bioinformatics*, Vol. 16, no. 10, pp. 906-914, 2000.

[8] X. Zhou and K.Z. Mao, "LS Bound based gene selection for DNA microarray data," *Bioinformatics*, Vol. 21, no. 8, pp. 1559-1564, 2005.

[9] Y. Lee and C.-K. Lee, "Classification of multiple cancer types by multicategory support vector machines using gene express data," *Bioinformatics*, vol. 19, no. 9, pp. 1132-1139, 2003.

[10] N. Pochet, et al., "Systematic benchmarking of microarray data classification: assessing the role of non-linearity and dimensionality reduction," *Bioinformatics*, vol. 20, no.17, pp. 3185-3195, 2004.

[11] T. Li, C. Zhang, and M. Ogihara, "A Comparative Study of feature Selection and Multiclass Classification Method for Tissue Classification Based on Gene Expression," *Bioinformatics*, vol. 20, no. 15, pp. 2429–2437, 2004.

[12] D. Komura, et. al., "Multidimesional support vector machines for visualization of gene expression data," *Bioinformatics*, vol. 21, no.4, pp. 439-444, 2005.

[13] A. Statnikov, C.F. Aliferis, I. Tsamardinos, D. Hardin and S. Levy, "A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis," *Bioinformatics*, vol. 21, no. 5, pp. 631-643, 2005.

[14] S. Dudoit, J. Fridlyand and T.P. Speed, "Comparison of discrimination methods for the classification of tumors using gene expression data," *Journal of the American Statistical Association*, 97, pp. 77-87, 2002.

[15] M. Detting and P. Buhlmann, "Boosting for tumor classification with gene expression data," *Bioinformatics*, vol. 19, no. 9, pp. 1061-1069, 2003.

[16] W. Li and Y. Yang, "How many genes are needed for a discriminant microarray data analysis," *Methods of Microarray Data Analysis*, S.M. Lin, and K.F. Johnson, Kluwer Academic, pp. 137-150, 2002.

[17] K.Y. Yeung, R.E. Bumgarner and A.E. Raftery, "Bayesian model averaging: Development of an improved multiclass, gene selection and classification tool for microarray data," *Bioinformatics*, vol. 21, no. 10, 2394-2402, 2005.

[18] S.-Y. Ho, L.-S. Shu and J.-H. Chen, "Intelligent Evolutionary Algorithms for Large Parameter Optimization Problems," *IEEE Trans. Evolutionary Computation*, vol. 8, no. 6, pp. 522-541, Dec. 2004.

[19] S.-Y. Ho, C.-H. Hsieh, H.-M. Chen and H.-L. Huang, "Interpretable gene expression classifier with an accurate and compact fuzzy rule base for microarray data analysis," *BioSystems*, vol. 85, pp. 165-176, 2006.

[20] H.-L. Huang, C.-C. Lee and S.-Y. Ho, "Selecting a minimal number of relevant genes from microarray data to design accurate tissue classifiers," *BioSystems*, 2006. (in press)

[21] S.-Y. Ho, H.-M. Chen, S.-J. Ho and T.-K. Chen, "Design of Accurate Classifiers with a Compact Fuzzy-Rule Base Using an Evolutionary Scatter Partition of Feature Space," *IEEE Trans. Systems, Man, and Cybernetics—Part B*, vol. 34, no. 2, pp. 1031-1044, April. 2004.

[22] S.-Y. Ho and Y.-C. Chen, "An efficient evolutionary algorithm for accurate polygonal approximation," *Pattern Recognition*, vol. 34, no. 12, pp. 2305-2317, 2001.

[23] S.-Y. Ho, C.-C. Liu and S. Liu, "Design of an optimal nearest neighbor classifier using an intelligent genetic algorithm," *Pattern Recognition Letter,* 23, pp. 1495-1503, 2002.

[24] K. Huang and R.F. Murphy, "Boosting accuracy of automated

classification of fluorescence microscope images for location proteomics," *BMC Bioinformatics*, vol. 5, no.78, 2004.
[25] T.G. Dietterich, "Ensemble methods in machine learning," *Lecture Notes in Computer Science Volume 1857*, Springer-Verlag; 1-15, 2000.
[26] J. Kittler and K. Messer, "Fusion of multiple experts in multimodal biometric personal identity verification systems," *IEEE International Workshop on Neural Networks for Signal Processing (NNSP12)* pp. 3-12, 2002.
[27] B. Boser, I. Guyon and V. Vapnik, "A training algorithm for optimum margin classifiers," *In the Fifth Annual Workshop on Computational Learning Theory*, Pittsburgh. ACM. pp. 144-152, 1992.
[28] C. Chang and C. Lin, *A library for support vector machines*, Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm
[29] S. Knerr, L. Personnaz and G. Dreyfus, "Single layer learning revisited: a stepwise procedure for building and training a neural network," *Neurocomputing: Algorithms*, Architectures and Applications. J. Fogelman (Ed.), Springer-Verlag, 1990.
[30] A.I. Belousov, S.A. Verzakov and J.V. Frese "A flexible classification approach with optimal generalization performance: Support vector machines," *Chemometrices and Intelligent Laboratory Systems*, 64, 15-25, 2002.
[31] M. Stone, "Cross-validatory choice and assessment of statistical predictions," *Journal of the Royal Statistical Society*, 36, pp. 111-147, 1974.
[32] H.-C. Kim, S. Pang, H.-M. Je, D. Kim and S.Y. Bang, "Constructing support vector machine ensemble," *Pattern Recognition*, 36, pp. 2757-2767, 2003.
[33] S.A. Vinterbo, E.Y. Kim and L. Ohno-Machado, "Small, fuzzy and interpretable gene expression based classifiers," *Bioinformatics*, 21, pp. 1964-1970, 2005.

TABLE I
THE FIVE DATASETS OBTAINED FROM THE WORK [13].

No.	Data set	Descriptions	# of classes	# of samples	# of genes
1	Brain_Tumor1	5 human brain tumor types	5	90	5920
2	Brain_Tumor2	4 malignant glioma types	4	50	10367
3	Prostate_Tumor	Prostate tumor and normal tissue	2	102	10509
4	Leukemia1	Acute myelogenous leukemia (AML), Acute lympboblastic leukemia (ALL) B-cell, and ALL T-cell	3	72	5327
5	Leukemia2	AML, ALL, and mixed-lineage leukemia (MLL)	3	72	11225

TABLE II
THE LOOCV ACCURACIES AND NUMBERS OF USED GENES FOR BESVM AND NON-GENE-SELECTION CLASSIFIER MC-SVM. THE RESULTS OF MC-SVM ARE OBTAINED FROM THE WORK [13].

Data set	# of genes in MC-SVM	MC-SVM (%)	BESVM (%)	# of genes in BESVM
Brain_Tumor1	5920	91.67	94.00	12.03
Brain_Tumor2	10367	77.00	85.00	15.25
Prostate_Tumor	10509	92.00	95.10	7.06
Leukemia1	5327	97.50	98.61	7.48
Leukemia2	11225	97.32	98.61	8.57
Mean	8669.6	91.10	94.26	10.08

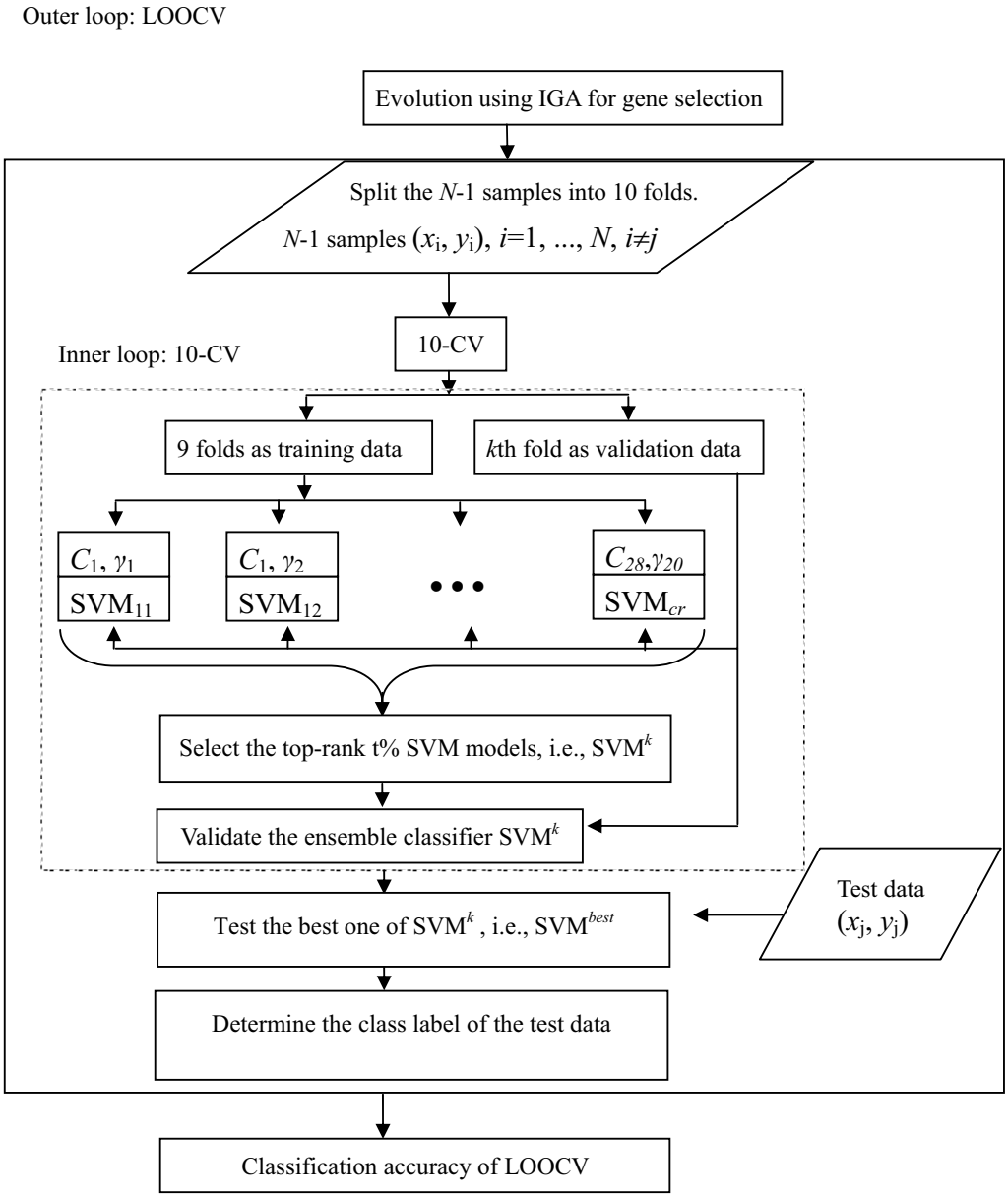

Fig. 2. Illustration of outer and inner loops related to the boosting evolutionary SVM classifier.